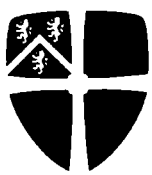
Topics in statistics of spatial-temporal disease modelling

Richardson, Jennifer

Topics in statistics of spatial-temporal disease modelling

Jennifer Richardson

A Thesis presented for the degree of
Doctor of Philosophy



Statistics and Probability Group
Department of Mathematical Sciences
Durham University
England

July 2009

Dedicated to

My family

Topics in statistics of spatial-temporal disease modelling

Jennifer Richardson

Submitted for the degree of Doctor of Philosophy
July 2009

Abstract

This thesis is concerned with providing further statistical development in the area of space-time modelling with particular application to disease data. We briefly consider the non-Bayesian approaches of empirical mode decomposition and generalised linear modelling for analysing space-time data, but our main focus is on the increasingly popular Bayesian hierarchical approach and topics surrounding that. We begin by introducing the hierarchical Poisson regression model of Mugglin *et al.* [36] and a data set provided by NHS Direct which will be used to illustrate our results throughout the remainder of the thesis. We provide details of how a Bayesian analysis can be performed using Markov chain Monte Carlo (MCMC) via the software `LinBUGS` then go on to consider two particular issues associated with such analyses. Firstly, a problem with the efficiency of MCMC for the Poisson regression model is likely to be due to the presence of non-standard conditional distributions. We develop and test the 'improved auxiliary mixture sampling' method which introduces auxiliary variables to the conditional distribution in such a way that it becomes multivariate Normal and an efficient block Gibbs sampling scheme can be used to simulate from it. Secondly, since MCMC allows modelling of such complexity, inputs such as priors can only be elicited in a casual way thereby increasing the need to check how sensitive our output is to changes to the prior. We therefore develop and test the 'marginal sensitivity' method which, using only one MCMC output sample, quantifies how sensitive the marginal posterior distributions are to changes to prior parameters.

Declaration

The work in this thesis is based on research carried out in the Statistics and Probability Group, Department of Mathematical Sciences, Durham University. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

I'd firstly like to thank my supervisor, Peter Craig for his continual help and support throughout my PhD especially during the final stages when my circumstances were more difficult...I really appreciate everything. I'd also like to thank 'the guys from the office' Anna, Gina, Dave, Mark, Lisa, Gav, Brett and Alicia for their friendship and especially for all the coffee breaks and lunches. A big thank you also to my Parents and Parents-in-law who have all supported and helped me in so many ways. I just started to name a few but there are so many I don't know where to start...thanks for everything. I'd also like to thank my Grandparents who have encouraged and believed in me throughout and my sister Steph too for keeping me sane, taking me shopping and never asking about my work! I also owe a massive thank you to Mark for his constant support, encouragement and so much more. Finally, thanks to our beautiful daughter Sophie for just being her!

Contents

Abstract iii

Declaration iv

Acknowledgements v

1 Introduction 1

1.1 Space-time disease modelling 1

1.2 Markov chain Monte Carlo 5

1.2.1 Efficiency 6

1.2.2 Prior Sensitivity 8

1.3 Thesis outline 9

1.4 New contributions 11

2 Analysis of a space-time DHF data set 12

2.1 DHF data set 13

2.1.1 Spatial representation of the data 14

2.2 Empirical Mode Decomposition 18

2.2.1 The sifting process 18

2.3 Analysis of DHF data using EMD 22

2.3.1 EMD of Bangkok incidence data 22

2.3.2 3-year periodic IMFs 22

2.3.3 Role of Bangkok in DHF dynamics 25

2.4 Generalised Linear Modelling of DHF data 28

2.4.1 DHF data frame 29

2.4.2 Model 1 . . . 30
2.4.3 Model 2 . . . 31
2.4.4 Model 3 . . . 32

3 Bayesian analysis of NHS Direct data using LinBUGS 37
3.1 NHS Direct data . . . 38
3.1.1 Data collection . . . 38
3.1.2 Exploring the data . . . 39
3.2 The model . . . 43
3.3 Using LinBUGS . . . 45
3.3.1 The script file . . . 45
3.3.2 Specifying the model . . . 46
3.3.3 Specifying the data . . . 48
3.3.4 Initial values . . . 50
3.3.5 Output . . . 51
3.4 Interpreting the results . . . 54
3.4.1 Variables of interest . . . 54
3.4.2 Posterior densities . . . 55
3.4.3 Representations of the posterior relative risk . . . 57
3.4.4 Is the NHS Direct data reliable? . . . 61
3.5 Model Assessment . . . 61

4 Improving the efficiency of MCMC 63
4.1 Motivation . . . 63
4.1.1 Simplified example . . . 64
4.2 Current research in this area . . . 65
4.3 Generalisation of the auxiliary variable method . . . 66
4.3.1 Scaling . . . 67
4.3.2 Location shift . . . 67
4.4 Adapting the auxiliary variable method . . . 68
4.4.1 Requirements for $\psi(\cdot)$ and $g(x)$ . . . 68
4.4.2 The plausible range of x values . . . 69

4.4.3 Choosing $\psi(\cdot)$ and $g(x)$ . . . 75
4.4.4 The problem with the method . . . 77
4.5 Auxiliary mixture approximation . . . 82
4.5.1 Univariate case . . . 84
4.5.2 Multivariate case . . . 91
4.6 Properties of the multivariate auxiliary approximation method . . . 93
4.6.1 Simulated examples . . . 94
4.6.2 Checking that posterior contains true value . . . 94
4.6.3 Mahalanobis distance . . . 97
4.6.4 Variance decomposition . . . 99
4.6.5 The role of the **y** values . . . 100

5 Prior sensitivity analysis of MCMC output 105
5.1 Background . . . 105
5.1.1 A current sensitivity method . . . 106
5.1.2 Relative entropy . . . 108
5.2 Marginal sensitivity method . . . 110
5.2.1 Notation . . . 110
5.2.2 Relative entropy of marginal posteriors . . . 111
5.2.3 Kernel density estimation . . . 112
5.2.4 Importance sampling . . . 112
5.2.5 Weighted Kernel density estimation . . . 113
5.2.6 Numerical integration . . . 113
5.2.7 Summary . . . 114
5.2.8 Graphical representation of sensitivity . . . 114
5.3 How well does this method work? . . . 115
5.3.1 True and estimated relative entropy for a Normal distribution 116
5.3.2 True and estimated relative entropy for a Gamma distribution 118
5.3.3 How well does importance sampling work? . . . 119
5.3.4 Importance sampling diagnostics . . . 120
5.3.5 Effect of importance sampling on the relative entropy estimate 124
5.3.6 Further exploration of statistical properties . . . 125

5.3.7 Method as a screening measure . . . 128
5.4 Replacing relative entropy with Kolmogorov distance . . . 135
5.4.1 Implementing the method . . . 136
5.4.2 How the method performs . . . 137
5.4.3 Other metrics . . . 139
5.5 Application . . . 139
5.5.1 MCMC sample from baseline prior . . . 139
5.5.2 Marginal sensitivity . . . 140
5.5.3 Sensitivity of the full posterior . . . 145
5.6 Marginal sensitivity analysis of BUGS output . . . 148
5.6.1 Necessary information . . . 148
5.6.2 `WinBUGS` power plant pumps example . . . 149
5.6.3 `WinBUGS` rats example . . . 150
5.6.4 Summary . . . 152

6 Conclusion **153**
6.1 Analysis of DHF data . . . 153
6.2 Bayesian analysis of NHS data . . . 154
6.3 Improved auxiliary sampling method . . . 154
6.4 Marginal sensitivity method . . . 155

Bibliography **158**

Appendix **164**

A NHS Direct Data **164**

List of Figures

2.1 Map of Thailand 13
2.2 Monthly DHF incidence for all provinces 15
2.3 DHF incidence rates for each year 16
2.4 DHF incidence rates for each month 17
2.5 EMD example 19
2.6 Sifting process example 20
2.7 EMD of Bangkok time series data 23
2.8 3rd IMF for all provinces 24
2.9 3rd IMF for northern provinces 24
2.10 3rd IMF for southern provinces 24
2.11 CCFs between 3-yr IMF of Bangkok and all other provinces 26
2.12 CCFs between 3-yr IMF of Bangkok and northern provinces 26
2.13 Percentage of the population living in an urban area 28
2.14 Residuals of Model 1 30
2.15 Month effect 32
2.16 Spatial effect 33
2.17 Year effect 34
2.18 Residuals of Model 2 35
2.19 Residuals of Model 3 36

3.1 Number of calls by symptom 40
3.2 Spatial structure of the cough data 41
3.3 Spatial structure of the cough data scaled by population 42
3.4 Traceplots resulting when two different sets of initial values are used . 51

3.5 Traceplots used as an informal check of convergence 52
3.6 Autocorrelation functions before thinning 53
3.7 Autocorrelation functions after thinning 54
3.8 Posterior densities 56
3.9 Average s_{it} for grouped time periods 58
3.10 s_{it} over time for each PCT 59
3.11 s_{it} over time for PCTs grouped by northern or southern 60
3.12 Spatial structure of the initial behaviour of the temporal plots 60

4.1 Illustrating the plausible range of x 70
4.2 Illustrating an upper bound for e^x 73
4.3 Illustrating the problem with the method 78
4.4 Illustrating the behaviour of -log(x) and -cosh(x) 80
4.5 Illustrating the behaviour of $-exp\{x\}$ and $\alpha(x^2 + x)$ 82
4.6 Univariate example showing accuracy of the approximation 87
4.7 Breakdown of the auxiliary mixture approximation for large y 88
4.8 $exp\{x - e^x\}$ and its Normal mixture approximation 88
4.9 Improved auxiliary mixture approximation for large y 90
4.10 Accuracy of the approximation for the multivariate case 104

5.1 Changes to the Normal distribution measured by relative entropy . . 109
5.2 Changes to the Gamma distribution measured by relative entropy . . 110
5.3 Prior change against marginal posterior change 115
5.4 True versus estimated relative entropy for Normal distribution 118
5.5 True versus estimated relative entropy for Gamma distribution 119
5.6 Importance sampling examples using the $\sum_{i=1}^{n} w_*(x_i)^2$ diagnostic . . 122
5.7 True relative entropy versus the $\left| \frac{\sum_{i=1}^{n} w(x_i)}{n} - 1 \right|$ diagnostic 123
5.8 $N = 50, n = 0, \bar{x} = 0$ 129
5.9 $N = 500, n = 0, \bar{x} = 0$ 129
5.10 $N = 3000, n = 0, \bar{x} = 0$ 130
5.11 $N = 3000, n = 2, \bar{x} = 0$ 130
5.12 $N = 3000, n = 30, \bar{x} = 0$ 131

5.13 $N = 3000, n = 1000, \bar{x} = 0$ 131
5.14 $N = 3000, n = 2, \bar{x} = 1$ 132
5.15 $N = 3000, n = 30, \bar{x} = 1$ 132
5.16 $N = 3000, n = 1000, \bar{x} = 1$ 133
5.17 $N = 3000, n = 2, \bar{x} = 2$ 133
5.18 $N = 3000, n = 30, \bar{x} = 2$ 134
5.19 $N = 3000, n = 1000, \bar{x} = 2$ 134
5.20 Changes to the Normal distribution measured by Kolmogorov distance 135
5.21 Changes to the Gamma distribution measured by Kolmogorov distance 136
5.22 Relationship between relative entropy and Kolmogorov distance . . . 137
5.23 True versus estimated Kolmogorov distance for Normal distribution . 138
5.24 True versus estimated Kolmogorov distance for Gamma distribution . 138
5.25 Marginal sensitivity to changes to hyperparameters a and b 141
5.26 Marginal sensitivity to changes to hyperparameters μ_{β_0} and τ_{β_0} 141
5.27 Marginal sensitivity to changes to hyperparameters μ_{β_1} and τ_{β_1} 142
5.28 Marginal sensitivity to changes to hyperparameters μ_{β_2} and τ_{β_2} 142
5.29 Marginal sensitivity to changes to hyperparameters μ_{θ_0} and τ_{θ_0} 143
5.30 Marginal sensitivity to changes to hyperparameters μ_{θ_1} and τ_{θ_1} 143
5.31 Marginal sensitivity to changes to hyperparameters μ_{θ_2} and τ_{θ_2} 144
5.32 Summary of marginal sensitivity for prior change of 0.2 144
5.33 Summary of marginal sensitivity for prior change of 0.2 (using Kolmogorov distance) 146
5.34 'Full sensitivity' for prior change of 0.2 146
5.35 pumps DAG 149
5.36 rats DAG 151

List of Tables

2.1 DHF incidence data 14

2.2 DHF Data frame used for generalised linear modelling 29

2.3 Summary of standard errors for month, province and year coefficients 31

2.4 Analysis of deviance for Model 3 34

3.1 Prior and posterior quantiles 55

4.1 Parameter values for the 10-component Normal mixture approximation 85

4.2 Summary values for the univariate example 86

4.3 Differences in cross-correlations obtained using the two approaches . . 93

4.4 p-values resulting from the Chi-squared test for uniformity 96

4.5 Mahalanobis distance between true values and prior 97

4.6 Mahalanobis distance between true values and samples 98

4.7 Quantiles of d_0 in $d_1, \ldots, d_{10000}$ 98

4.8 V for data case when $\mu_i = 1$ 101

4.9 V_1 for data case when $\mu_i = 1$ 101

4.10 V_2 for data case when $\mu_i = 1$ 102

4.11 $V - (V_1 + V_2)$ for data case when $\mu_i = 1$ 102

4.12 Summary of y values 103

5.1 Twelve combinations of N, n and $\bar{x}$ 126

A.1 Number of cough calls in each PCT per week 164

Chapter 1

Introduction

1.1 Space-time disease modelling

Infectious diseases are frequently dominating news headlines and there is an increasing need to understand their epidemic behaviour. Statistical analyses have many uses, for example providing a descriptive picture of the epidemic, identifying areas of particular risk or looking at the impact of interventions such as vaccination. Gaining further insight into these things is extremely useful in health planning and allocating of resources to combat such diseases. Spatial-temporal statistical modelling of both infectious and non-infectious diseases is an active research area, in both human health and veterinary medicine. There is a substantial literature which is reviewed in Lawson [29] and Ashby [1] and the following examples highlight important aspects of this literature.

Lung cancer is an example of a non-infectious disease in human health for which one space-time data set in particular has been extensively studied, namely rates in the 88 counties of Ohio during the period 1968-88. The following authors have used this dataset to help further develop spatial-temporal models. Waller *et al.* [50] extend existing spatial models to account for temporal effects and spatio-temporal interactions. However, time is essentially treated as exchangeable so there is less emphasis on modelling the temporal development of the disease risk. Also, by their own admission, their results are difficult to interpret since they don't take into account some important factors such as smoking prevalence. Knorr-Held and Besag [25]

also note that their results are not ideal due to the absence of direct information on smoking and other important county effects, but they describe approaches that adjust for these unmeasured covariates. Their model combines existing models for longitudinal and spatial data in a hierarchical Bayesian framework with particular emphasis on the role of space- and time- varying coefficients. However they combine temporal and spatial main effects additively, they do not allow for space×time interactions. Knorr-Held [26] extends this work and compares four models that have different space×time interaction terms. A simple measure of urbanisation is also incorporated as a surrogate for cigarette consumption and other risk factors associated with urban areas.

Turning our attention to infectious diseases now, we first consider advances in veterinary medicine. Lawson and Zhou [30] discuss various issues around modelling foot and mouth disease and apply a descriptive space-time model to UK data from the 2001 epidemic. This is essentially a binomial model with various random effect terms which estimates well certain patterns of the disease. However, they note that additional terms could be included which would further mimic the infectious behaviour. This, along with many existing models, is mainly concerned with the disease behaviour on a large scale, such as country level. However, the very recent paper by Picado *et al.* [37] explores the use of a space-time interaction tool as an indicator of local behaviour. It therefore provides useful insight for future model developments.

While there is definite interest in space-time modelling of infectious diseases within veterinary medicine, there has been considerably more within the human health field. Examples include work by Mugglin *et al.* [36] who develop a space-time model which they use to analyse an influenza data set. Their model incorporates the Bayesian hierarchical technology previously used for modelling non-infectious diseases (for example by [25, 50] for modelling lung cancer). This work provides a smoothed and interpretable description of what happened during the epidemic and the approach is well suited to non-sparse infectious disease data where there are clearly distinguishable epidemic curves. Knorr-Held and Richardson [27] present a similar model which is suitable for sparse data with small increases and apply it to a

meningococcal disease data set. With this particular disease, there is evidence that it has short term increases superimposed from time to time onto the overall epidemic, which are known as hyperendemic states. These are of particular public health interest therefore a central feature of this model is the possibility to calculate, for each region and time point, the probability of being in one of these states. Another recent development in this area is work by Chiogna and Gaetan [6] who develop a descriptive space-time model for the behaviour of a measles epidemic. They adopt a formulation based on the Kriged Kalman Filter model of Mardia *et al.* [33] which is extended to deal with count data. The approach is quite general and can easily be refined so is likely to be useful for the study of other infectious diseases.

We have named just a few examples but there has been a continual development of statistical methodology in this field over recent years. This has been due to its usefulness in health planning, disease surveillance and intervention, and allocating health funding. The availability of data with which to develop such statistical models can often pose a problem in human health due to reasons of confidentiality. One way around this is for the data to be presented as counts aggregated in space and time so no individual can be identified. Count data has therefore become one of the most widely used data types in disease modelling.

Focussing now on applications to human infectious diseases, there are a number of different directions for studies. A few examples are relative risk assessment, cluster detection and surveillance. In relative risk assessment we are interested in finding regions and/or times of excess risk of the disease. Usually the data will reflect a population background effect as well as the excess risk and this background effect is often represented by an expected number of cases within a region and time period. Frequently used models for assessing relative risk are of the form $y_{it} \sim Poisson(E_{it}\theta_{it})$ for infinite populations or a Binomial equivalent for finite populations. Here, y_{it} is the disease count, E_{it} is the expected number of cases and θ_{it} is the relative risk associated with region i and time period t. Mugglin *et al.* [36] is an example of this type of study. They use a Bayesian approach and interpret the posterior relative risks for an influenza data set. Another direction for study is that of cluster detection which is a topic of great public health interest and involves

assessing where and when clusters of disease occur. Clusters are usually thought of as an unusual aggregation of excess risk in local areas of a geographic region. There are a range of cluster detection methods available and a number of reviews of them exist such as Diggle [12] and Lawson and Kulldorff [28]. Many studies either focus on purely spatial cluster models or on modelling spatial-temporal patterns of diseases without directly modelling spatial-temporal clustering. A recent paper by Yan and Clayton [51] attempts to fill the gap. They extend a purely spatial cluster model to accommodate space-time clustering using a Bayesian framework. One further direction for study is surveillance. Surveillance systems collect and monitor data for disease trends and outbreaks which is of considerable public health interest. The object of statistical surveillance is to detect a change in a disease process accurately and quickly as new observations are added. Rodeiro and Lawson [41] discuss methodological issues in developing a quick response in surveillance systems. They consider some exploratory statistical methods as well as more sophisticated ones based on hierarchical space-time models.

As well as a range of directions for studies, there are also range of approaches to modelling the data. One approach is to use a descriptive model which doesn't include any information about transmission or incubation for the disease but generally provides a smoothed and interpretable description of what happened during the epidemic. Examples of these types of models can be found in Mugglin *et al.* [36] and Knorr-Held and Richardson [27] which we have already discussed. An alternative type of model includes some form of transmission dynamic and generally splits the population into groups such as susceptible, exposed, infective and removed. One example is the model of Le Menach *et al.* [31]. They focus on foot and mouth disease and build a stochastic model at farm level where initially each farm is classified as susceptible, then moves through various stages until the animals are culled and the farm becomes 'removed'.

We have already mentioned a few articles that use a Bayesian hierarchical approach to modelling, namely [25, 27, 36, 50]. This approach has become increasingly popular over recent years due to its very flexible framework which allows extremely complicated models to be built out of a succession of relatively simple components.

Observable outcomes are modelled conditionally on certain parameters which themselves are given a probabilistic specification in terms of further parameters, known as hyperparameters. If need be, these hyperparameters can then be given further probabilistic specifications, and so on. A non-hierarchical approach can be inappropriate for some data. For example, models involving few parameters generally cannot fit large data sets accurately, but if they involve many parameters they tend to fit the existing data well but lead to bad predictions for new data (known as overfitting). Conversely, a hierarchical model can have enough parameters to fit the data well and the population distribution can structure some dependence into the parameters thereby avoiding problems of overfitting. These models can be evaluated using Markov chain Monte Carlo (MCMC) methods which have attracted much attention over recent years.

This thesis is concerned with count data of infectious diseases within the human health field. We concentrate on descriptive modelling and focus mainly on topics surrounding the Bayesian hierarchical approach using MCMC, although some non-Bayesian approaches to modelling are also briefly considered.

1.2 Markov chain Monte Carlo

MCMC methods are a class of algorithms for sampling from multidimensional probability distributions that are difficult to sample from directly. They are generally used to sample from the posterior distribution of a complex Bayesian model. Brooks [4] provides a comprehensive review of some of the most common areas of research in this field and Gilks *et al.* [20] provides numerous examples on the use of MCMC methods. The algorithms are based on constructing a Markov chain which has the desired posterior distribution as its stationary distribution. Examples of MCMC methods include Gibbs sampling and Metropolis-Hastings algorithms.

Gibbs sampling involves building a Markov chain whose dependence on the predecessor is controlled by the conditional distributions. It involves simulation from the distribution of each parameter in turn conditional on the most recent values of all other parameters available. A common approach to this is to group the parameters

into a number of blocks and simulate from the joint conditional distribution of each block of parameters given the most recent values of all other parameters available. This can be very beneficial computationally especially when parameters are highly correlated and is of most benefit when the blocks are large. Gibbs sampling is a very popular method as it doesn't require any tuning (i.e. preliminary MCMC runs in order to establish reliable values for certain parameters) however it does require that all conditional distributions are of standard form. It can often be difficult to implement if the required conditional distributions assume awkward forms. In such cases we may turn to the Metropolis-Hastings algorithm.

Metropolis-Hastings has the advantage of being able to provide a solution when the conditional distributions are complex. It involves proposing a candidate value randomly and then deciding whether or not to keep it as the next value in the Markov chain. It can be quite difficult to propose good candidate values and can involve high computational effort but works well once it is tuned properly.

1.2.1 Efficiency

There are a number of issues to consider when using MCMC methods, one being the efficiency of the method. Since MCMC allows modelling of extremely complex models, it could take a long time to run and therefore can only produce a relatively small sample from the posterior. In cases such as this we would need to consider ways of improving the computational efficiency. There has been much interest in this issue over recent years, particularly in finding ways to generate from 'non-standard' conditional distributions using a Gibbs sampling approach.

Damien *et al.* [11] discuss an approach which, after the introduction of auxiliary variables, results in a Gibbs sampler having a set of easily sampled standard full conditionals. Suppose that we have a density $f(x) \propto l(x)\pi(x)$ where $l(x)$ is some non-negative function and $\pi(x)$ is a density. Suppose also that $f(x)$ is not possible to sample from directly. Then the general idea is to introduce a latent variable u and an extra full conditional for u in such a way that all but one of the full conditionals are uniform densities and the remaining one is a truncated version of π. This idea is applied in the context of Bayesian non-conjugate and hierarchical models. It has

the advantage of being easy to code since it requires only standard random variate generation routines. However, Damien *et al.* don't claim to improve efficiency in every case and note that a Metropolis-Hastings algorithm may well be preferable in some cases.

An alternative approach for tackling such problems is known as auxiliary mixture sampling and has been an active research area over recent years. The most current version of the method is described in detail in section 4.5, but the approach basically involves introducing two sequences of auxiliary variables in such a way that a Gibbs sampling scheme can be used on models which otherwise would require an alternative. Frühwirth-Schnatter and Wagner [17] show how a Poisson regression model can be transformed into an approximate Normal linear model using these auxiliary variables. They introduce the first sequence as the unobserved inter-arrival times of the Poisson process. This eliminates the non-linearity in the observation equation but the error term is still non-Normal. They then approximate the error term by a mixture of Normal densities and introduce the second sequence of auxiliary variables as the component indicators of the mixture. A Gibbs sampling scheme for unknown quantities is then described which only requires random draws from standard distributions. Gschlöβl and Czado [21] extend this approach to spatial Poisson regression models and also compare the Gibbs sampling scheme with a Metropolis-Hastings approach. They conclude that the Metropolis-Hastings method requires more computational effort but the Gibbs sampling scheme needs to be run for considerably longer in order to obtain the same precision of the parameters. Frühwirth-Schnatter and Frühwirth [15] move away from Poisson regression and show that the method is feasible for models involving other discrete-valued observations such as binary and multinomial data. The second data augmentation step is essentially the same as that of the Poisson model but the first is different. The first step introduces the utility of choosing category 1 as auxiliary variables for binary data and the utilities of choosing categories 1 to m for multinomial data. However, a disadvantage of each of the above approaches is that the number of auxiliary variables introduced via the first sequence can be very high. For example, $y_i + 1$ latent variables are needed for each observation y_i in the Poisson model case. This means that the method is only

really useful for data with small counts. Frühwirth-Schnatter *et al.* [16] propose an improved version of auxiliary mixture sampling for count data, binomial data and multinomial data which involves a reduced number of latent variables. They introduce at most two auxiliary variables for each observation instead of $y_i + 1$ for the Poisson model, one instead of the number of repetitions N_i for binomial data and $m-1$ instead of $(m-1)N_i$ for multinomial data. They present two case studies in which the method allows them to approach large hierarchical models using block Gibbs sampling. However, the question arises as to whether the method can be improved to only introduce one sequence of auxiliary variables. This is considered further in chapter 4 of this thesis.

1.2.2 Prior Sensitivity

One further issue to consider is that since MCMC allows modelling of such complexity, inputs such as priors can only be elicited in a very casual way. This means that there is an increasing need to consider the sensitivity of output to changes in the model inputs. This is part of the wider issue of robust Bayesian analysis and a comprehensive overview of the main topics in this area is provided in Rios Insua and Ruggeri [40]. It begins with a review of the approach by Berger *et al.* [3] then goes on to deal with many issues surrounding the topic, including a number of case studies. It aims to give both researchers and practitioners an opportunity to become quickly and thoroughly acquainted with this field.

There has been much work focussing on local sensitivity where small changes to the prior are studied. One such example is Millar [35] who quantifies local sensitivity using derivatives and suggests a method for automating this during Bayesian model fitting in WinBUGS. However, attention here is restricted to estimating the derivatives of a summary measure $E(g(\theta) \mid y)$ where $g(\theta)$ denotes a function of the unknown parameters values. A further example is McCulloch [34] who develops a general method for assessing the influence of model assumptions in a Bayesian analysis. In particular he looks at the effect of changing the hyperparameter away from the initial choice and uses relative entropy to measure the difference between the posteriors resulting from different choices of hyperparameter. However, this

approach requires us to know the resulting posterior distributions which is not always possible. In the case where MCMC methods are used, we do not know the posterior distribution exactly but instead have only a sample from it. Clarke and Gustafson [7] (whose work extends the idea of McCulloch [34]) suggest how this method could be applied in the case where MCMC methods are used, but they do not pursue this. A drawback of the suggestion is that it doesn't allow us to see exactly which parts of the posterior are affected by the changes to the prior, which would be both interesting and useful to know. Gustafson [22] does address the issue of which parts of the posterior are affected. He considers how sensitive the marginal posterior distributions are to changes in various parts of the prior and uses derivative norms as measures of sensitivity. However, the focus here is on assessing the sensitivity of posterior expectations rather than the distribution as a whole. It seems what is needed is a method to bridge the gap. Using a metric such as relative entropy to measure the discrepancy between two marginal distributions would take into account other aspects of posterior distribution, not just the mean. This is considered further in chapter 5 of this thesis.

1.3 Thesis outline

This thesis is concerned with providing further statistical developments in the area of space-time modelling with particular application to disease data. While we do consider some non-Bayesian methods, the main focus is on the increasingly popular Bayesian hierarchical approach using MCMC. Chapters 2 and 3 are concerned with analysing two space-time data sets using both Bayesian and non-Bayesian methods, whereas the later chapters are concerned with providing statistical developments for use in the Bayesian context. Throughout this thesis we make use of the software R[1] which is a free environment for statistical computing and graphics. R also has a set of downloadable packages[2] written by many different authors, which we make use

[1] available from `http://www.stats.bris.ac.uk/R/`

[2] available from `http://CRAN.R-project.org/`

of. We also use OpenBUGS[3] (Bayesian inference Using Gibbs Sampling) which is a piece of computer software for the Bayesian analysis of complex statistical models using MCMC methods. It has been developing over the years and the latest version can run on Windows (known as WinBUGS [32]) and Linux (known as LinBUGS).

We begin chapter 2 with the analysis of a space-time data set on the number of cases of dengue haemorrhagic fever (DHF) in Thailand. We do not adopt the popular Bayesian approach to analysis here, but instead look more closely at a recent method adopted by Cummings *et al.* [10] using empirical mode decomposition (EMD) and also investigate the data further using generalised linear modelling (GLM). EMD is based on the idea that a complicated time series can essentially be thought of as a number of waves riding on top of each other. The method identifies these waves and decomposes the data into a finite number of intrinsic mode functions (IMFs) each representing a different characteristic timescale. EMD is very complex and not very clearly defined whereas GLM is easy to fit, clearly specified and we found it to recover much of the same information as EMD. This leads us to conclude that it is probably better to use one of the standard statistical models rather than EMD to analyse such data.

Chapter 3 is concerned with the analysis of a space-time data set provided by NHS Direct. It comprises the number of calls made to the north east site about the symptom cough. Here, we do adopt a Bayesian approach and analyse the data using the space-time hierarchical model of Mugglin *et al.* [36]. In the context of this example, we give details of how to implement a Bayesian analysis using the software LinBUGS and the R package CODA.

In chapter 4 we look at the issue of improving the efficiency of MCMC for Poisson regression models such as the one introduced in chapter 3. Such models involve at least one non-standard conditional distribution and our goal in this chapter is to find a way to make them take a standard form by augmentation and then present an efficient block Gibbs sampling scheme for sampling from them. We consider the possibility of using the methods introduced in section 1.2.1 to achieve this, the first

[3]available from http://mathstat.helsinki.fi/openbugs/

being the auxiliary variable method of Damien *et al.* [11]. We describe the approach in more detail and show how it doesn't appear to be a feasible option for our problem. We then consider the auxiliary mixture sampling method of Frühwirth-Schnatter *et al.* [16]. We first describe the method in more detail and then present an improved version of it which only involves one sequence of auxiliary variables. We illustrate this using a simplified version of the model and data introduced in chapter 3 and also examine how well the method works.

Chapter 5 is concerned with the prior sensitivity issue mentioned in section 1.2.2. We focus on the situation where we have one set of output from MCMC with which to analyse the sensitivity and restrict attention to where the prior distribution is the only model input to be changed. We first consider in more detail the method presented in Clarke and Gustafson [7] highlighting the drawback of the approach in that it doesn't allow us to see exactly which parts of the posterior are affected by the changes to the prior. We then develop the marginal sensitivity method which uses a similar approach to quantify how sensitive the posterior distribution of each parameter is to changes in the prior. We also examine how well the method works and consider how particular adaptations to it affect the results. We again illustrate this new method using the model and data introduced in chapter 3 and finish by explaining how we could go about producing a general piece of software for the marginal sensitivity analysis of BUGS output resulting from any model.

1.4 New contributions

The novel contributions of this thesis are as follows

- a Bayesian analysis of new NHS Direct space-time data found in chapter 3
- the 'improved auxiliary sampling' method introduced in chapter 4
- the 'marginal sensitivity' method introduced in chapter 5

Chapter 2

Analysis of a space-time DHF data set

Dengue fever is a disease found in the tropics which is transmitted to humans by a particular type of mosquito. Some of the cases can occur as the severe, life-threatening form of the disease known as dengue haemorrhagic fever (DHF). Cummings *et al.* [10] examine the spatial-temporal dynamics of DHF incidence in Thailand using the method of empirical mode decomposition (EMD) introduced by Huang *et al.* [23]. They observe a three-year periodic travelling wave which originates in the capital, Bangkok and moves radially. A map of Thailand is shown in Figure 2.1 with the position Bangkok highlighted in orange.

Two common alternatives to EMD for analysing time series data are Fourier analysis and wavelets. EMD has the advantage over these methods of being able to handle both nonlinear and non-stationary signals. Furthermore, both Fourier analysis and wavelet methods use an underlying function which is fixed and does not necessarily match the varying nature of the signal, whereas EMD uses the signal itself with no underlying function. However, a disadvantage of EMD over these methods is that it is lacking a theoretical foundation and involves a number of ad hoc judgements when implementing it.

In this chapter we consider in more depth what is involved in the method of EMD and how it has been implemented for a particular DHF data set. We recreate the results presented in [10] and develop some of the thoughts further. We also use

Figure 2.1: Map of Thailand

a different method, namely generalised linear modelling, to gain further insight into the behaviour of DHF. We begin by introducing the data.

2.1 DHF data set

Numbers of DHF cases are routinely collected by the Ministry of Health in Thailand and this is available for the years 1983 to 1997 on the John Hopkins Centre for Immunisation Research website[1]. The data set is presented as the monthly number of cases of DHF per 1000 population for each of the 72 provinces of Thailand. Data for 5 provinces and the first 13 months are shown in Table 2.1.

[1] `http://www.jhsph.edu/cir/dengue.html`

Table 2.1: DHF incidence data

	MaeHongSon	ChiangMai	ChiangRai	Lamphun	Lampang
1983	0	0	0	0	0.01
1983.08	0	0	0	0	0.02
1983.17	0	0	0	0	0.02
1983.25	0	0	0	0	0
1983.33	0	0	0	0	0.01
1983.42	0	0.03	0.02	0.08	0.04
1983.5	0	0.19	0.18	0.35	0.09
1983.58	0.01	0.3	0.19	0.56	0.21
1983.67	0	0.12	0.16	0.56	0.3
1983.75	0.02	0.01	0.02	0.03	0.14
1983.83	0.02	0	0.01	0	0.03
1983.92	0.01	0	0.01	0	0.01
1984	0	0	0	0	0

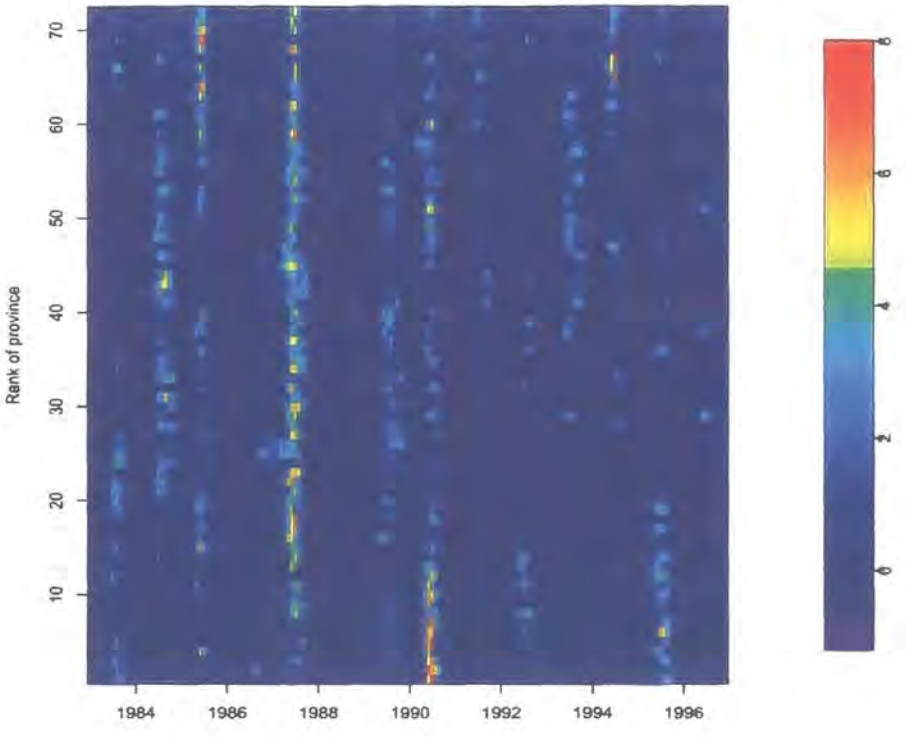
Cummings *et al.* [10] present log-transformed and normalised monthly incidence data for all years and all provinces. The command `image.plot()` from the R package `fields` was used to recreate their image which is shown in Figure 2.2. The provinces are arranged from the most southerly to the most northerly from bottom to top and the scale of the legend to the right of the image is the logarithm of cases per 100,000 people per month. The vertical lines evident in the image suggest that peaks in incidence occur at roughly the same time across all the provinces.

2.1.1 Spatial representation of the data

In order to get a further idea of what this data looks like, the R package `RArcInfo` was used to produce the following spatial representations. Figure 2.3 shows the average incidence rates of DHF per 1000 population each year and Figure 2.4 shows the same for each calendar month.

It is difficult to see any obvious pattern as we move from one year to the next in 2.3. It may be that a further breakdown into months is necessary for this to be

Figure 2.2: Monthly DHF incidence for all provinces



the case. However, it is clear which of the years experience high rates of incidence (shown in red/orange) and that the rates do vary between years.

It is clear from Figure 2.4 that the summer months have higher incidence rates than the others and that rates are very low in most provinces for the months November through to April. Also, it seems to be the provinces nearest to the northern border of Thailand that experience the highest incidence rates. It is interesting to note that Bangkok and the provinces around it don't seem to reach any of the high incidence rates, even in the summer months. This will be considered further in section 2.3.3. For now we turn our attention to the analysis of this data presented by [10] and first introduce the method of EMD.

Figure 2.3: DHF incidence rates for each year

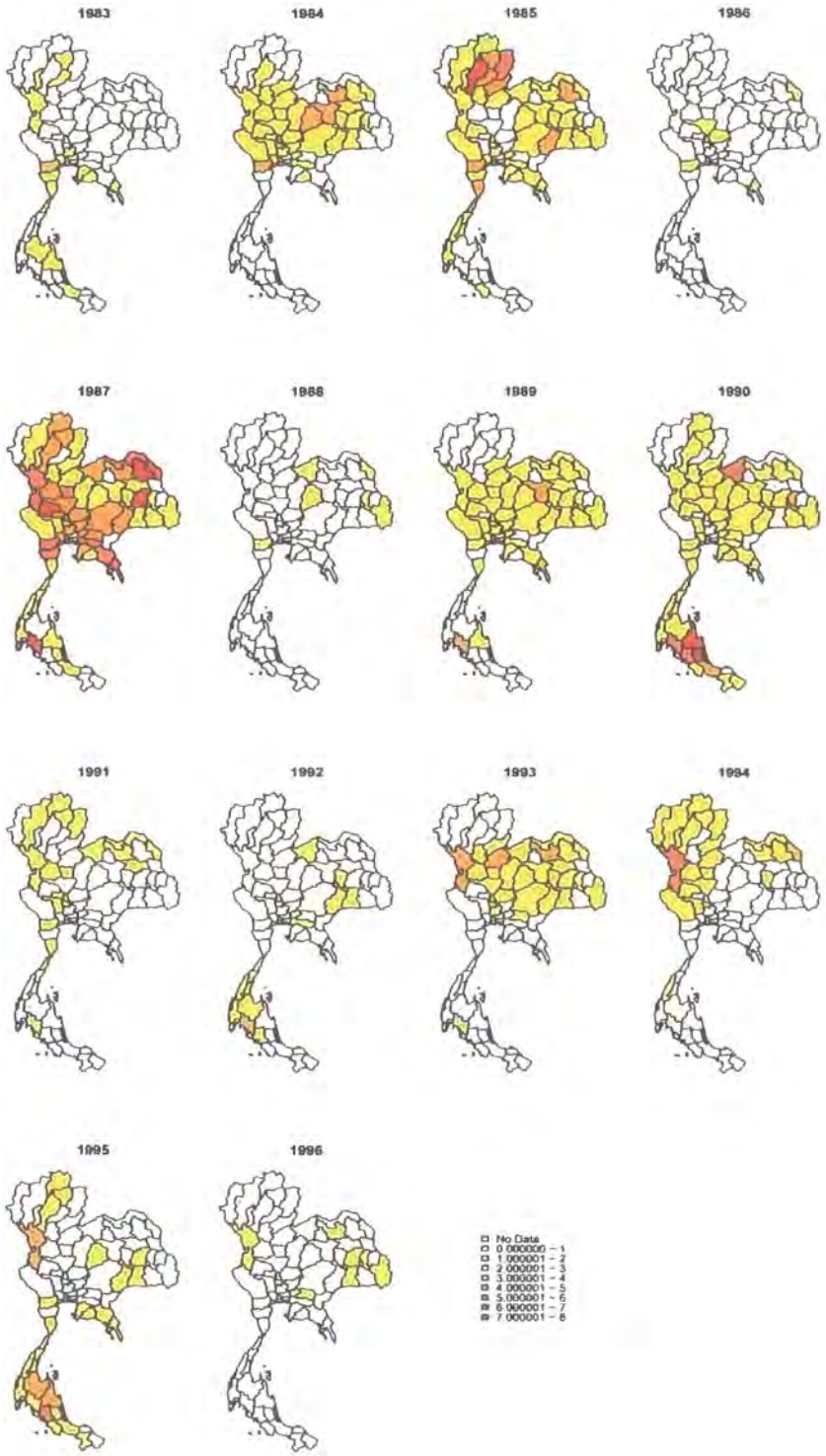
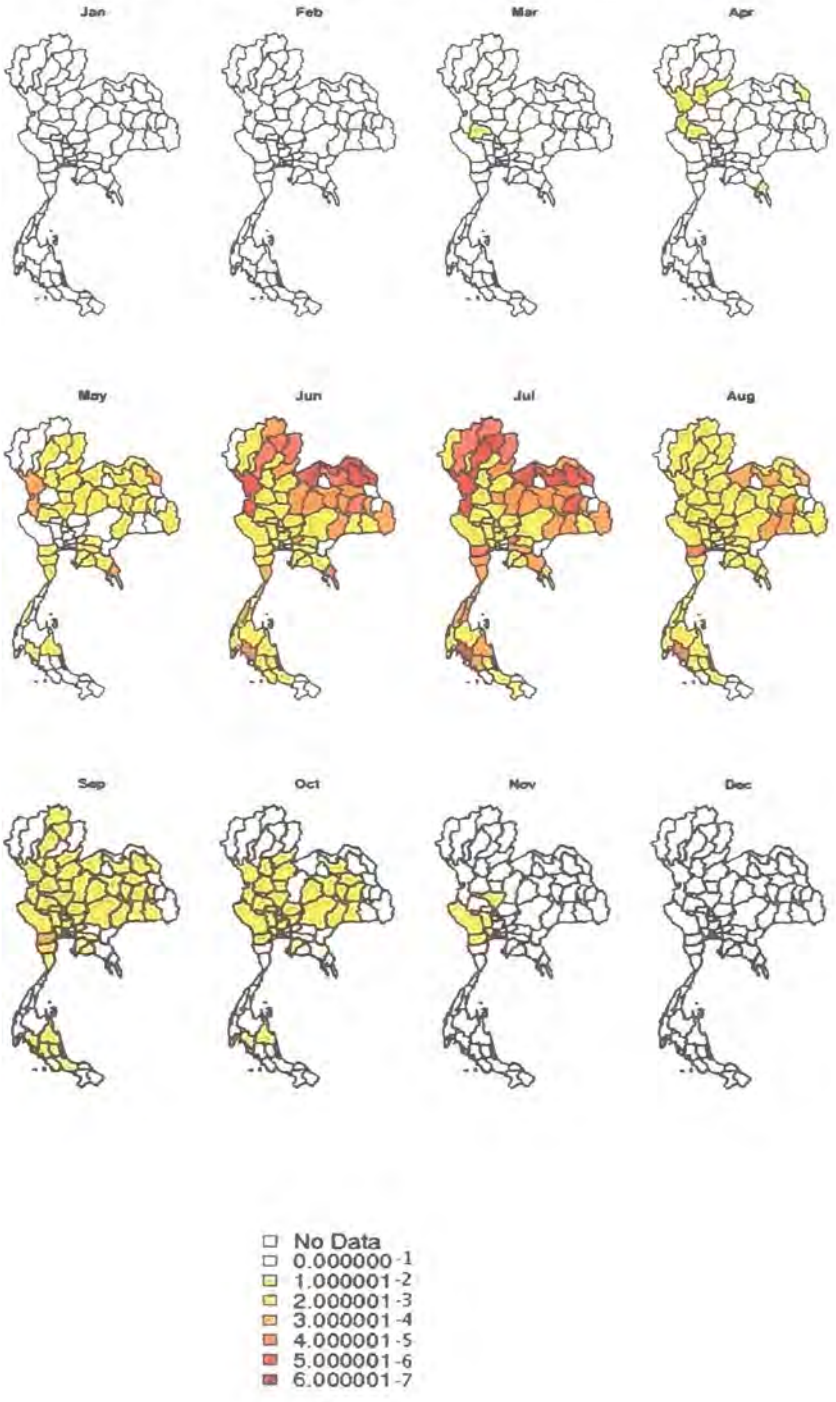


Figure 2.4: DHF incidence rates for each month



2.2 Empirical Mode Decomposition

EMD was introduced by Huang *et al.* [23] as a new method for analysing nonlinear and nonstationary time series data. Time series data can be complicated and this can be seen in the first series in Figure 2.5, the signal. Interlaced local extrema and zero crossings can be seen as well as negative local maxima and positive local minima. This suggests that the series involves a number of waves riding on top of each other. Each wave defines a characteristic timescale of the data and is intrinsic to the process. The characteristic timescale is defined by the time lapse between successive extrema. The method of EMD identifies these waves empirically and then decomposes the data accordingly into a finite number of intrinsic mode functions (IMFs). The middle three series in Figure 2.5 are the IMFs which were extracted from the signal by EMD. The characteristic timescales of all three of these can be seen in the original complicated signal. The original series is the sum of the IMFs extracted plus a residue. This is the series that remains once all the IMFs have been extracted and it should be either the mean trend of the data or a constant. The final series in Figure 2.5 is the residue.

2.2.1 The sifting process

EMD decomposes the time series into IMFs by means of a sifting process which is described as follows:

1. Identify the local maxima and local minima of a raw time series, $x(t)$.
2. Fit 2 cubic splines, one connecting the maxima and one connecting the minima, to form upper and lower envelopes with all the data between them, $e_{max}(t)$ and $e_{min}(t)$.
3. Calculate the mean of the 2 envelopes, $m(t) = \frac{e_{max}(t)+e_{min}(t)}{2}$.
4. Find the difference between the raw and the mean time series, $h(t) = x(t) - m(t)$.
5. Find out if $h(t)$ satisfies the following two conditions:

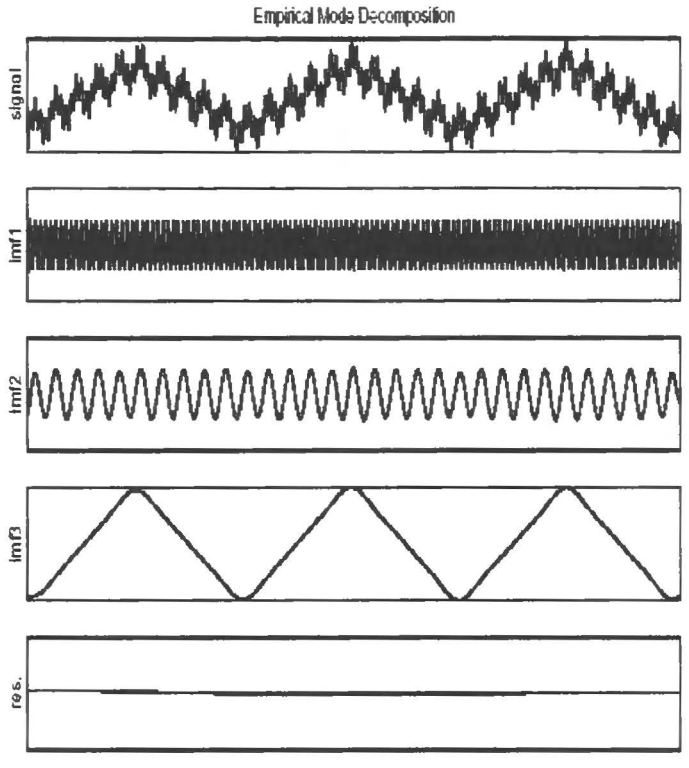


Figure 2.5: Example of EMD taken from Rilling *et al.* [39]

- The number of extrema and the number of zero crossings of $h(t)$ must not differ by more than 1. This will ensure that all local minima are negative and all local maxima are positive.
- The mean series connecting the cubic splines of the extrema of $h(t)$ must be zero at all times.

6. If $h(t)$ *does not* satisfy the above criteria then the algorithm is repeated using $h(t)$ as the raw series.

7. If $h(t)$ *does* satisfy the above criteria then it is the first IMF. It should then be subtracted from the raw series and the algorithm repeated on this difference to identify subsequent IMFs.

8. The sifting process is ended when no more IMFs can be extracted, i.e. no more than one local maxima or minima remains.

Figure 2.6 shows an illustration of this sifting process.

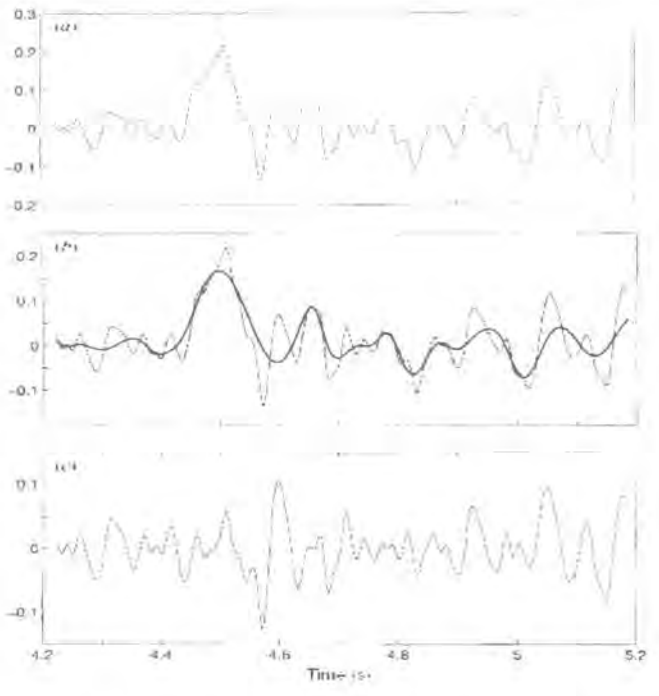


Figure 2.6: Example of the sifting process taken from Huang *et al.* [23]

(a) shows the raw time series, $x(t)$.

(b) shows the raw time series in the thin solid line, the upper and lower envelopes, $e_{max}(t)$ and $e_{min}(t)$, in the dashed lines and the mean, $m(t)$, in the thick solid line.

(c) shows $h(t)$, the difference between the data and the mean. This is the result after one sifting but it is not an IMF as negative local maxima and positive local minima can be seen suggesting riding waves. The algorithm will then need to be repeated using this $h(t)$ as the raw series.

Previous work using EMD has highlighted a number of issues to take into consideration when using the method, one being how to best fit the cubic spline of the extrema near the ends and another being when to stop the sifting algorithm. Rilling *et al.* [39] addressed both of these matters. They wrote m-function files for `MATLAB` and gave reference to a website[2] from which they could be downloaded. Here we use

[2] `http://perso.ens-lyon.fr/patrick.flandrin/software.html`

one such file, namely `emd.m`, which computes EMD according to Huang *et al.* [23] and incorporates the variations reported in Rilling *et al.* [39]. By looking in detail at the MATLAB commands used in this file, we are able to gain a better understanding of how exactly these two issues have been addressed.

Before fitting the cubic spline, the signal is extended at both ends by mirroring the extrema. This is achieved by fitting an imaginary line of symmetry vertically through the the first and the last extrema and mirroring two maxima and two minima at either end. This means that the signal is extended by eight extrema, four at either end.

In `emd.m`, the sifting process is kept going until one of the following is true:

- the number of extrema is less than 3, i.e. the series is just a constant trend since each end point is an extrema;
- 2000 iterations have been performed.

Rilling *et al.* [39] also introduced new stopping criteria for the process based on 2 thresholds, θ_1 and θ_2. Their aim was to guarantee *globally* small fluctuations in the mean while taking into account *locally* large excursions. They introduced the *mode amplitude*

$$a(t) = \frac{e_{max}(t) - e_{min}(t)}{2}$$

and the *evaluation function*

$$\alpha(t) = \left| \frac{m(t)}{a(t)} \right|.$$

The idea here is that the sifting is iterated until $\alpha(t) < \theta_1$ for some fraction, $(1 - \alpha)$ of the total duration, while $\alpha(t) < \theta_2$ for the remaining fraction. In `emd.m` the following default values are set: $\alpha = 0.05$, $\theta_1 = 0.05$ and $\theta_2 = 0.5$. This means that for 95% of the time, the evaluation function $\alpha(t)$ is restricted to values less than 0.05 but for 5% of the time it can take values of up to 0.5. This is making allowances for a small number of deviations from the wave.

2.3 Analysis of DHF data using EMD

In this section we look more closely at the results presented in [10]. They first performed EMD on the logarithm of the incidence data for each of the provinces which we illustrate in section 2.3.1 for the province of Bangkok. They then chose to focus on the 3-year periodic IMF because they find that it accounts for 44% of the interannual variability in dengue incidence. We present these 3-year IMFs for each province in Figure 2.8. Since they have reason to believe that Bangkok may have a central role to play in the dynamics of DHF in Thailand, they then look at the cross-correlation functions (CCFs) between the 3-year IMF for Bangkok and each of the other provinces. We again recreate these findings and present them in Figure 2.11.

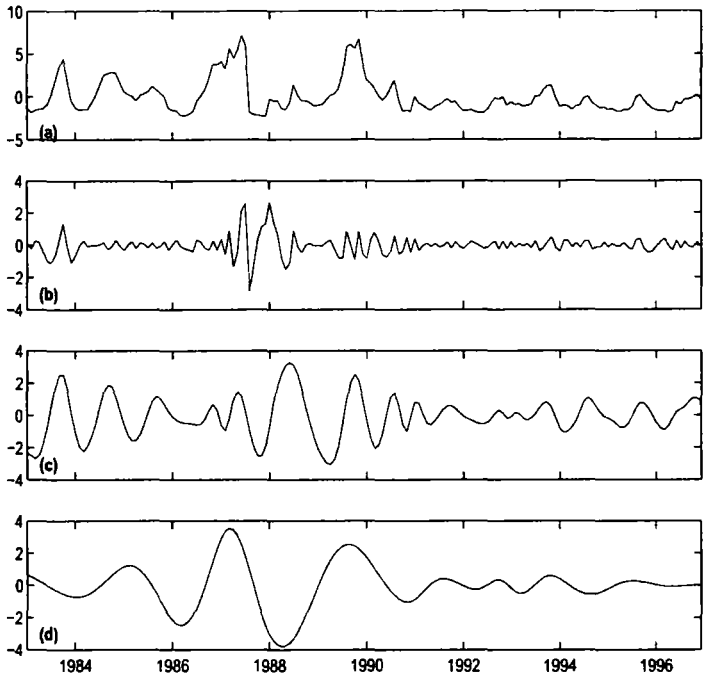
2.3.1 EMD of Bangkok incidence data

Here we use `emd.m` in `MATLAB` to decompose the incidence data for the province of Bangkok using the sifting process. Figure 2.7 shows the time series of the monthly incidence as well as the first three IMFs extracted.
(a) shows the monthly incidence of DHF cases in the province of Bangkok for the years 1983-1997 in which the complexity of the series is clear. The overall shape has peaks which have local maxima and minima evident within them (known as riding waves), the years 1987 and 1989 for example. These local extrema represent another characteristic timescale which is apparent in the first IMF extracted, shown in (b). As well as eliminating such riding waves, the sifting process also serves the purpose of making the wave profiles more symmetric. (c) shows the seasonal IMF and (d) shows the 3-yr periodic IMF both of which are obscured in the raw incidence data by the presence of many periodic components.

2.3.2 3-year periodic IMFs

Here we use `emd.m` to decompose the incidence data for each province and extract the 3-year periodic IMF. These are then shown in Figure 2.8 for each province arranged from the most southerly to the most northerly from bottom to top.

Figure 2.7: EMD of Bangkok time series data



Two patterns are clear in the image, one for the most southerly provinces (ranked 1 to 14) and a second for the remaining provinces. We can see from the map on page 13 that the 14 most southerly provinces could be thought of as a peninsula coming from the main body of Thailand which may account for why they have a slightly different pattern. Although Cummings *et al.* [10] didn't make any reference to a difference in DHF incidence or spread between southern and northern provinces, Figure 2.8 suggests there may be a distinction. To investigate this idea further we separate this into 3-yr periodic IMFs for the northern provinces and 3-yr periodic IMFs for the southern ones. These are shown in Figures 2.9 and 2.10 respectively.

Considering Figure 2.9 first, three vertical lines representing high rates of DHF incidence can be seen and the one at 1987 shows peaks across most of these northern provinces. Although this line can be seen to some extent in Figure 2.10, the three main vertical lines representing high rates start roughly at the same time as the last one in Figure 2.9 ends. This suggests that peaks in DHF incidence occur later in the southern provinces than they do in the rest.

Figure 2.8: 3rd IMF for all provinces

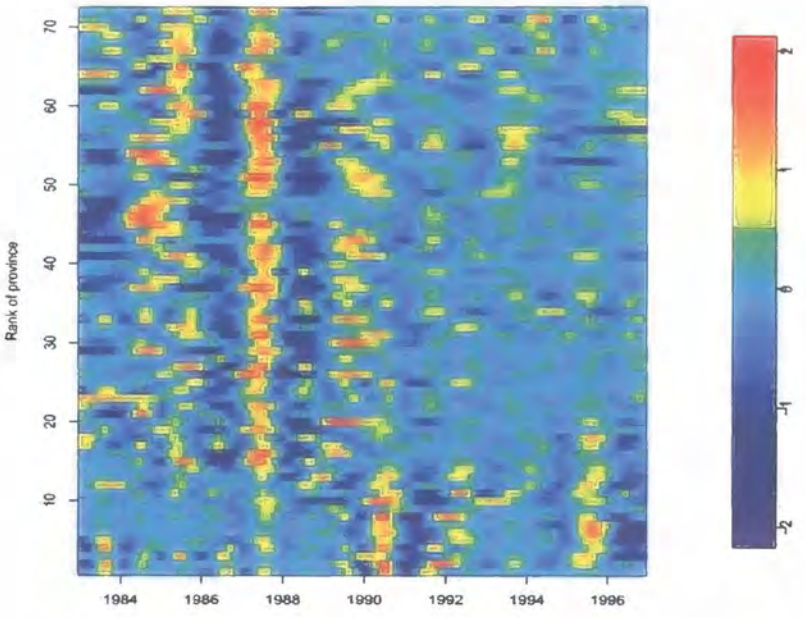


Figure 2.9: 3rd IMF for northern provinces

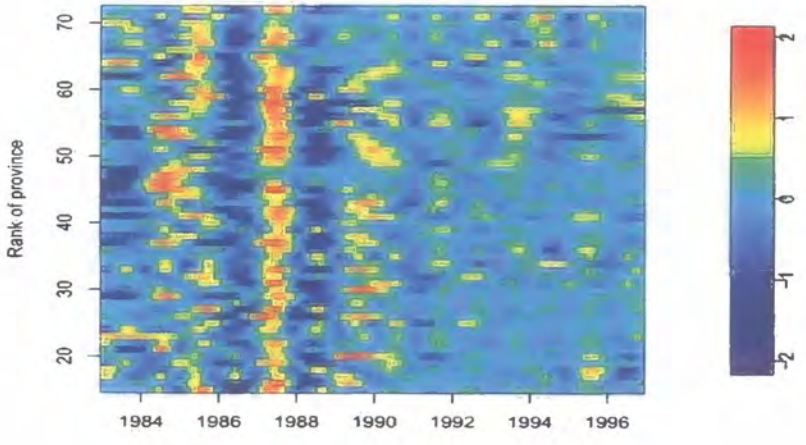
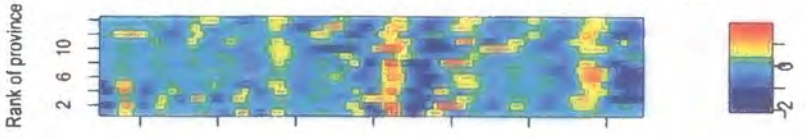


Figure 2.10: 3rd IMF for southern provinces



2.3.3 Role of Bangkok in DHF dynamics

Due to the size of Bangkok's population and its central role in the commerce of the country, Cummings *et al.* [10] examined its role in this 3-year travelling wave. To repeat their results we calculate the CCFs between the 3-year IMF for Bangkok and each of the other provinces using Pearson correlation coefficients. Figure 2.11 shows these for lags of -20 to 20 months with the provinces ordered from bottom to top with increasing distance from Bangkok. The negative numbers are for the case where Bangkok lags behind the province and the positive are for the province

Figure 2.11: CCFs between 3-yr IMF of Bangkok and all other provinces

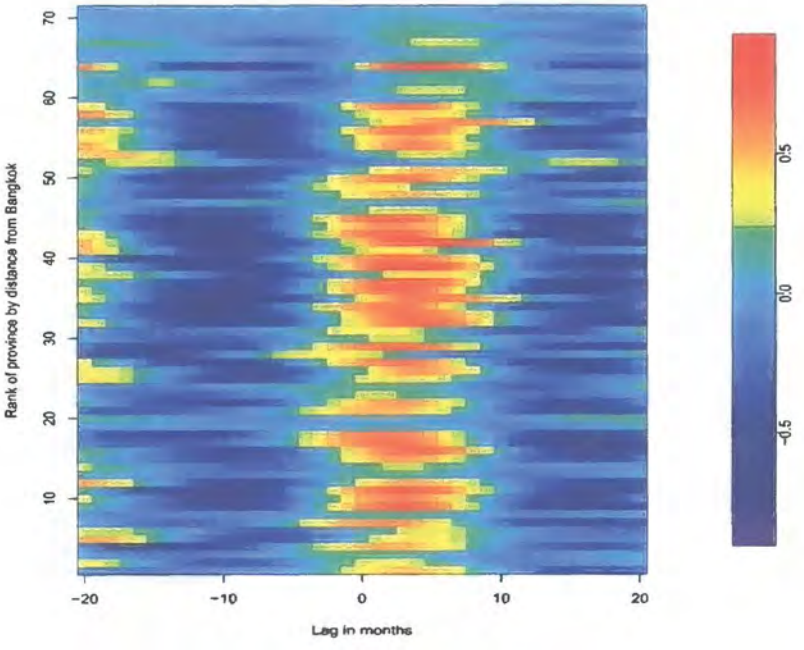
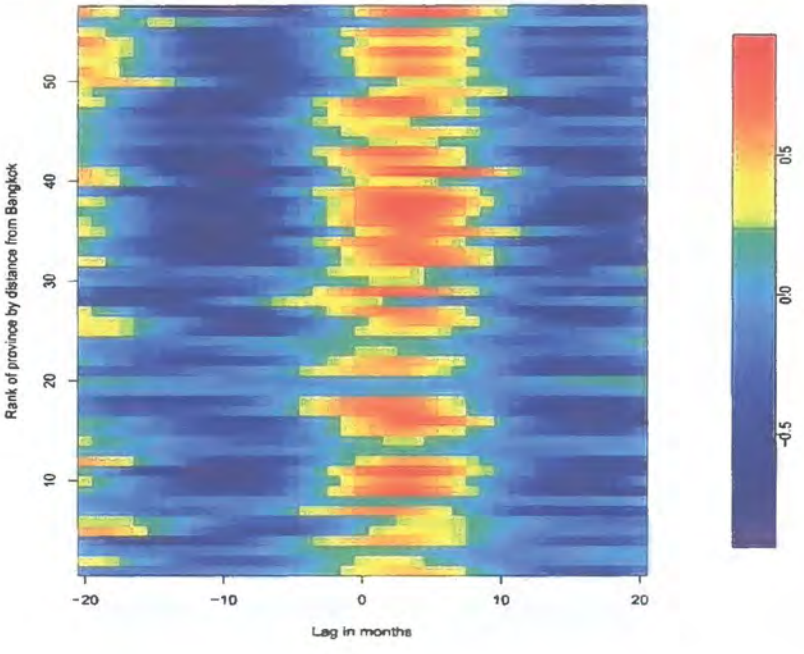


Figure 2.12: CCFs between 3-yr IMF of Bangkok and northern provinces



lagging behind Bangkok.

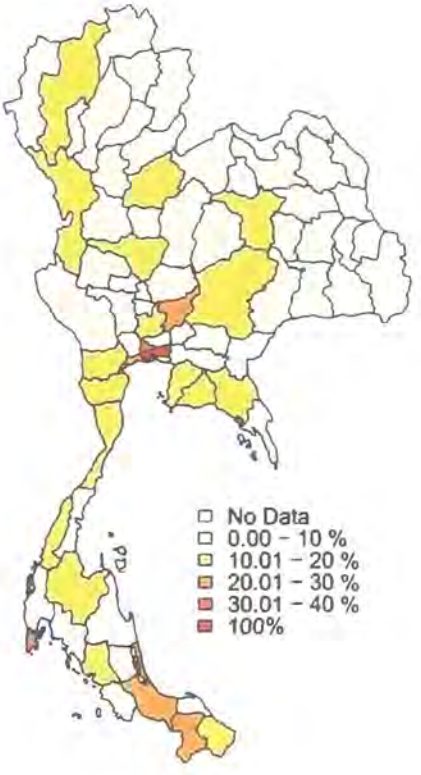
A vertical band of red can be seen in Figure 2.11, just to the right of the centre. This represents a strong positive correlation between the 3-yr periodic IMF of Bangkok and the same mode of each of the other provinces when the provinces lag behind Bangkok by between 0 and 8 months. It is evident in the image that while a vertical band is present overall, a strong correlation is not present within this band for all provinces. For example, provinces ranked 19, 20, 62, 63 and from 65 upwards are shown to have a CCF of around zero. The question arises as to whether these zero correlations may be between Bangkok and the southern provinces leaving the strong correlations to be between Bangkok and northern provinces. To see if this is the case, we reproduce the image but this time omitting the CCFs for the 14 most southern provinces. The result is shown in Figure 2.12. Although the overall pattern looks the same in the two images, it is clear that there is a reduction in the number of provinces with zero correlation in Figure 2.12. This further confirms our suspicion that there may be a distinction between patterns in DHF behaviour for northern and for southern provinces.

Cummings *et al.* [10] claim that this red band slants off to the right representing a greater lag with distance from Bangkok and therefore conclude that the 3-year periodic travelling wave emanates from Bangkok. However, this doesn't seem to be very clear in either of the two images. We now investigate the role of Bangkok further.

Recall that the spatial representations of the DHF data set in section 2.1.1 show that Bangkok and the areas around it don't seem to reach any of the high incidence rates. Since Bangkok has a large population and is thought of as essentially urban, the question arises as to whether there could be any link between DHF incidence and the percentage of a province's population living in urban areas. Figure 2.13 shows these percentages which were calculated using population data available from the College of Population Studies, Chulalongkorn University website[3].

[3]http://www.chula.ac.th/college/cps/thaidata/thailand_data.html

Figure 2.13: Percentage of the population living in an urban area



This shows that the provinces which have the greatest percentage of people living in urban parts include Bangkok and those surrounding it as well as on the southern border of Thailand. Those provinces on the northern border, which showed high rates of incidence in Figures 2.3 and 2.4, have a low percentage of urban population. This indicates that there may be a negative relationship between the percentage of urban population and DHF disease incidence.

2.4 Generalised Linear Modelling of DHF data

An alternative way to investigate this space-time DHF data set is via a generalised linear model (GLM) which we do using the `glm()` function in `R`. GLMs are defined in terms of three components:

1. *a distribution function*: this is the distribution that the observations $\mathbf{Y}$ take and must be a member of the exponential family

2. *a linear predictor*: this is a linear combination of unknown parameters $\boldsymbol{\beta}$ with covariates $\mathbf{x}$ as their coefficients

3. *a link function*: this relates the mean of each Y_i to the linear predictor

2.4.1 DHF data frame

We begin by changing the form of the data from that shown in Table 2.1 to the data frame shown in Table 2.2.

Table 2.2: DHF Data frame used for generalised linear modelling

	Time	Rates	Prov	Year	Region	Month	Pop	Counts	CountsInt
860	12	0.03075	5	1983	S	Dec	930123	28.60287	29
861	12	0.00860	4	1983	S	Dec	467621	4.0254	4
862	12	0.03914	2	1983	S	Dec	291166	11.39795	11
863	12	0.00000	44	1983	N	Dec	1683798	0.00000	0
864	12	0.00406	48	1983	N	Dec	475068	1.93117	2
865	13	0.00000	71	1984	N	Jan	148282	0.00000	0
866	13	0.00000	69	1984	N	Jan	1252241	0.00000	0
867	13	0.00204	72	1984	N	Jan	976634	1.99720	2
868	13	0.00000	68	1984	N	Jan	410484	0.00000	0
869	13	0.00000	67	1984	N	Jan	392588	0.00000	0
870	13	0.00379	66	1984	N	Jan	730057	2.76886	3

The columns of interest to us here are :

- `CountsInt` which is the actual number of DHF cases (obtained by multiplying the rate with the population size/1000) rounded to the nearest integer. These will form the observations y_{st} for our model.

- `Prov`, `Month` and `Year` which give the province index, month and year associated with each count respectively. They will form the covariates $\mathbf{x}$ for our model.

- Pop which gives the population of the province in the year concerned. This will form an offset term in our model.

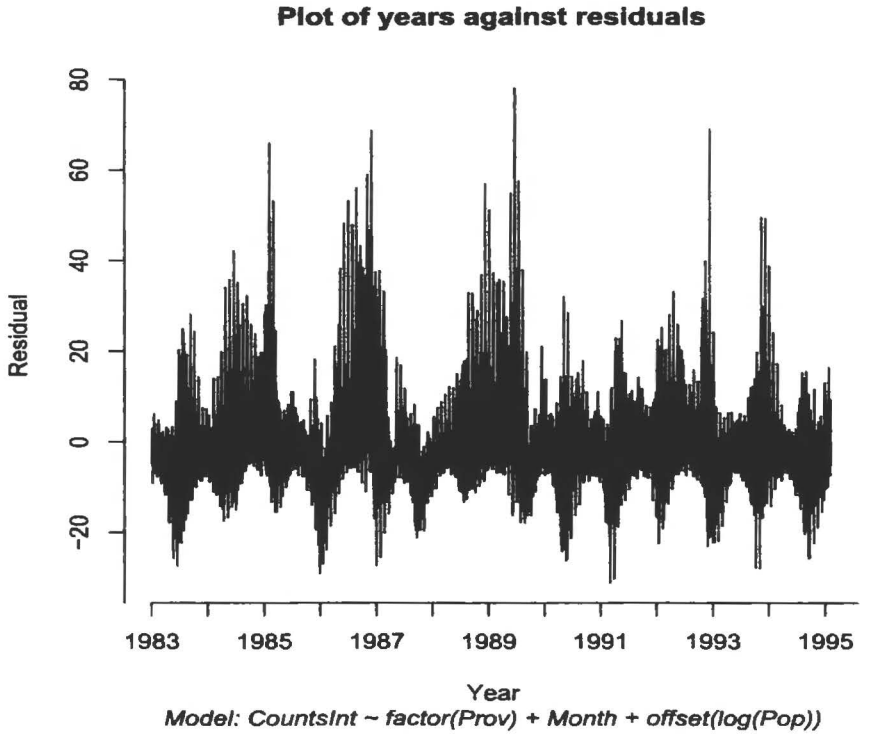
2.4.2 Model 1

The first model we consider is

$$\begin{aligned} Y_{st} &\sim Poisson(P_s\lambda_{st}) \\ log(P_s\lambda_{st}) &= \beta_s x_s + \beta_t x_t \end{aligned}$$

for $s = 1, \ldots, 72$ and $t = 1, \ldots, 12$ where Y_{st} is the number of DHF cases in province s and month t. P_s is the population of province s. x_s is the province covariate, x_t is the month covariate and β_s and β_t are the unknown parameters to be estimated from the data.

Figure 2.14: Residuals of Model 1



Residuals can be used to explore the adequacy of fit of a model. After fitting our model we look at the residuals which are plotted in Figure 2.14 against year.

They were calculated using the R command `resid()` which has the usual deviance residuals for GLM as its default. It seems from this that there may be a repeating pattern evident, increasing to a peak and then decreasing again every few years. We now adapt our model to include a year effect and in particular look for whether the parameter associated with it shows a 3-year repeating cycle as suggested by Cummings *et al.* [10].

2.4.3 Model 2

Now suppose that

$$\begin{aligned} Y_{stu} &\sim Poisson(P_{su}\lambda_{stu}) \\ log(P_{su}\lambda_{stu}) &= \beta_s x_s + \beta_t x_t + \beta_u x_u \end{aligned}$$

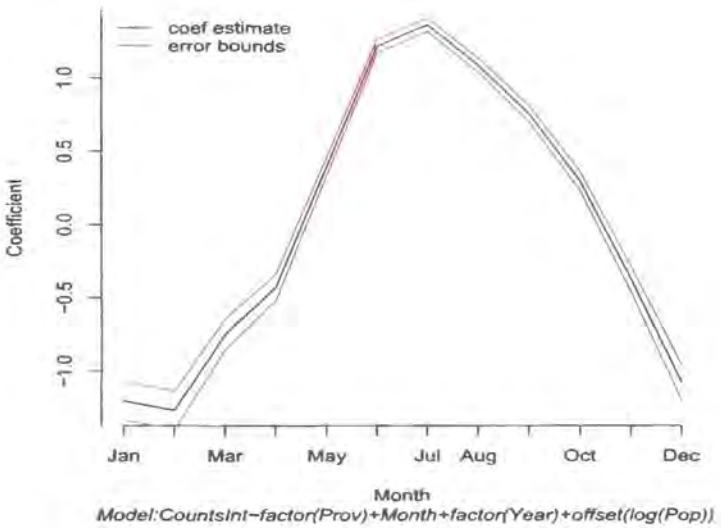
for $s = 1, \ldots, 72$, $t = 1, \ldots, 12$ and $u = 1, \ldots, 15$ (representing years 1983, ..., 1997). Using the `glm()` function in R we estimate the parameters β_s, β_t and β_u for all s, t and u. These are shown for province, month and year respectively in Figures 2.15, 2.16 and 2.17 along with their upper and lower error bounds. The upper and lower bounds consist of the coefficient estimate plus or minus twice the standard error. Since the model has signs of over-dispersion, these standard errors have been scaled up using the estimate of dispersion. These scaled standard errors for each set of coefficients are also summarised in Table 2.3.

Table 2.3: Summary of standard errors for month, province and year coefficients

	Month	Province	Year
Min.	0.0220	0.0394	0.0017
1st Qu.	0.0264	0.0684	0.0288
Median	0.0382	0.0848	0.0321
Mean	0.0410	0.0970	0.0325
3rd Qu.	0.0541	0.1169	0.0387
Max.	0.0665	0.2698	0.0475

These plots provide us with further insight into the data and therefore DHF dynamics. It is clear from Figure 2.15 that there is a seasonal effect evident which

Figure 2.15: Month effect



peaks in the summer months. This is consistent with the peak in incidence evident in Figure 2.4. Figure 2.16 shows clustering at the north-western border, around Bangkok and at the southern border. This could be thought of as consistent with the clustering of high percentage of urban population shown in Figure 2.13. It is difficult to see a clear pattern in Figure 2.17. There doesn't seem to be a definite 3-year repeated cycle here although the pattern evident for the 3-year periods 1983 to 1986 and 1988 to 1991 do look to be similar.

When we look at the residuals of Model 2, which are shown in Figure 2.18, we see that there is still some structure visible. We now consider a third model which adds in second order interactions to see if this structure in the residuals is removed.

2.4.4 Model 3

Now suppose that

$$\begin{aligned} Y_{stu} &\sim Poisson(P_{su}\lambda_{stu}) \\ log(P_{su}\lambda_{stu}) &= \beta_s x_s + \beta_t x_t + \beta_u x_u + \beta_{su} x_s x_u + \beta_{tu} x_t x_u + \beta_{st} x_s x_t \end{aligned}$$

for $s = 1, \ldots, 72$, $t = 1, \ldots, 12$ and $u = 1, \ldots, 15$ (representing years $1983, \ldots, 1997$). Using the `glm()` function in R we estimate all of the β coefficients for all s, t and

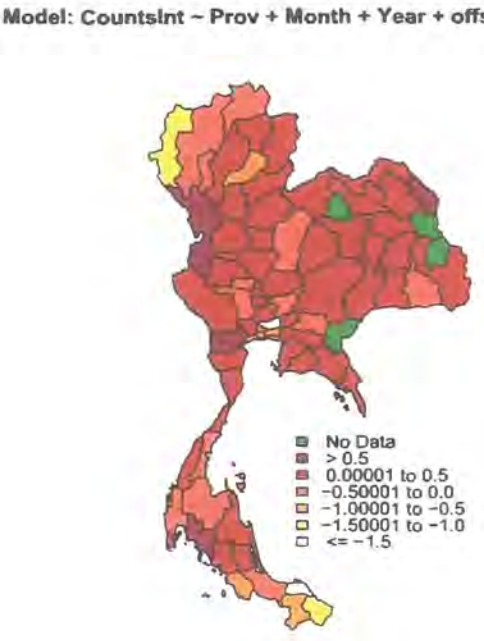


Figure 2.16: Spatial effect

u. Figure 2.19 gives the residuals for this model with the year boundaries shown in red. We can see that they look much better than those of Model 2 but there is still some strange behaviour for 1987 and also 1990 to a lesser extent. If we look again at Figure 2.2 on page 15 which shows the monthly incidence rates for the data broken down by year, we see that there are peaks in the disease rate across all provinces during 1987 and in the southern provinces during 1990. It therefore seems that this model doesn't cope well with high peaks in disease rate. The Analysis of Deviance for this model is shown in Table 2.4 and we can see that the most significant amount of variation is coming from the Month term.

The disadvantage of GLM is that it is a simple type of statistical model and a more complex model may be necessary to more accurately fit some types of data. However GLM is more appropriate for our data than EMD for a number of reasons.

Figure 2.17: Year effect

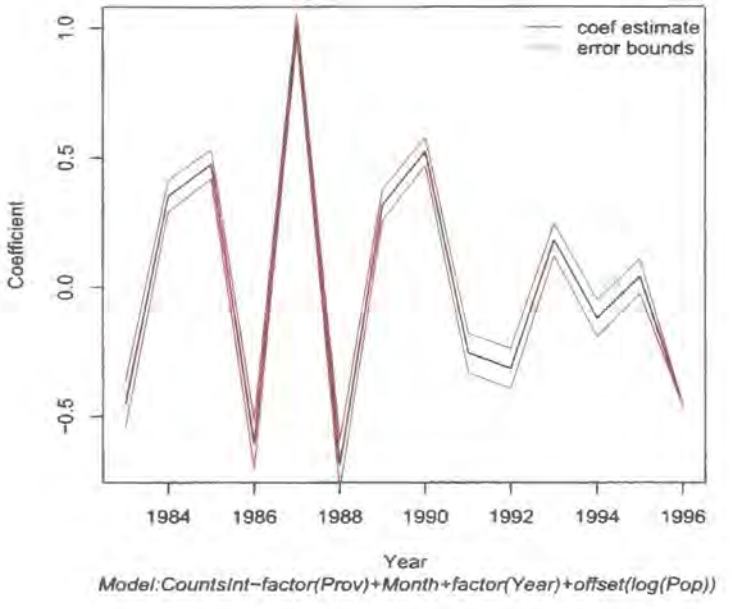
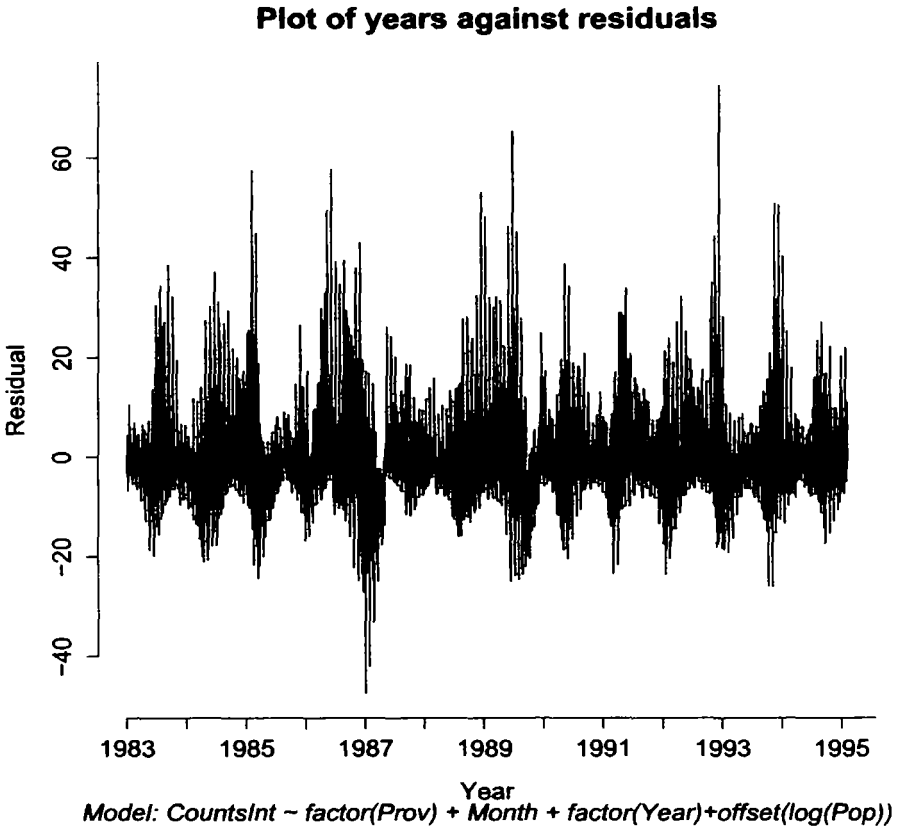


Table 2.4: Analysis of deviance for Model 3

	Df	Deviance Resid.	Df Resid.	Dev
NULL			12095	1468526
factor(Prov)	71	90751	12024	1377775
factor(Year)	13	204080	12011	1173695
factor(Month)	11	564665	12000	609030
factor(Prov):factor(Year)	923	264172	11077	344858
factor(Prov):factor(Month)	781	128707	10296	216151
factor(Year):factor(Month)	143	111942	10153	104209

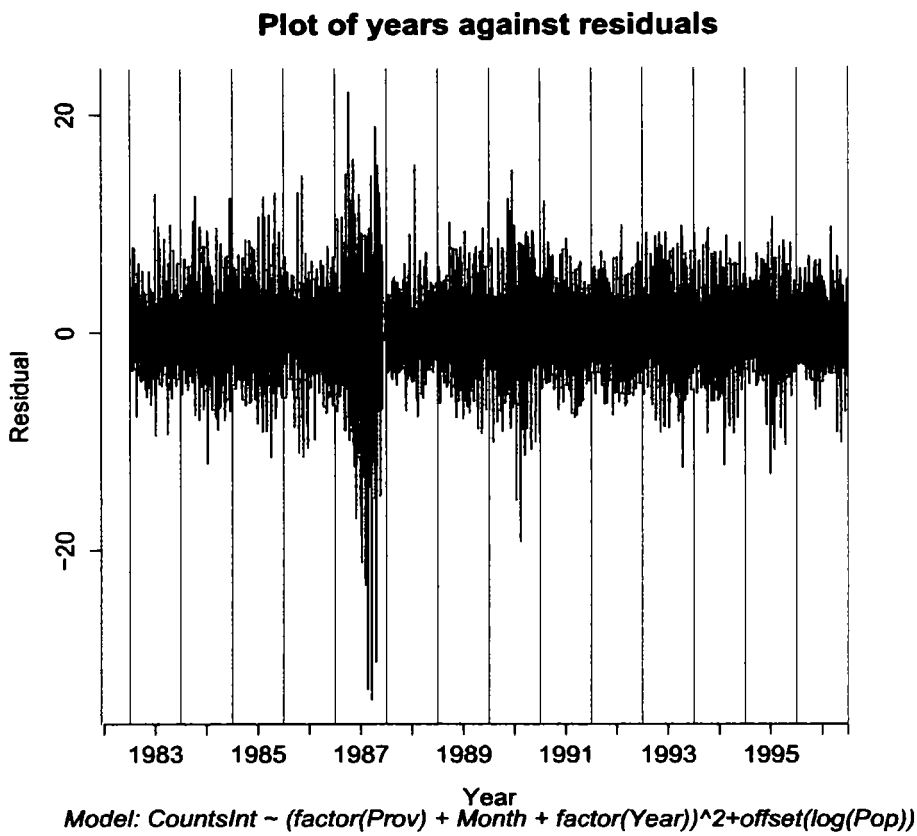
EMD is not a statistical model but is purely descriptive and therefore doesn't have any uncertainty associated with it. Furthermore, it isn't clear exactly how it works. A number of people have produced different computer code for it and each of the methods differ slightly. Two particular issues people differ on are how to best fit the cubic splines near the ends and when to stop the sifting algorithm. These were discussed in more detail in section 2.2.1. In contrast, GLM is a widely used statistical model and therefore the estimates come with standard errors, residuals and formal

Figure 2.18: Residuals of Model 2



statistical prodedures for comparing models. We can progressively increase the complexity of the model and check for improvement in fit, which is not possible with EMD. A model for the travelling wave idea considered in this chapter could be incorporated into GLM and checked for fit, whereas EMD is only able to take the data apart and explore the idea in an ad hoc way.

Figure 2.19: Residuals of Model 3



Chapter 3

Bayesian analysis of NHS Direct data using LinBUGS

One basic type of data that arise in space-time epidemiology is that of count data. This is where cases of disease are accumulated and the count is associated with a region (e.g. postcodes, electoral wards, counties) and a time period (e.g. day, week, month). For reasons of confidentiality this is one of the most common types of data available. In this chapter we explore count data provided by the NHS Direct north east site to see if there is any spatial structure in the spread of infection in the North East area.

NHS Direct is a national telephone helpline for health advice. A computerised database stores information about each call received and data extracted from this database will provide a timely snapshot of symptoms occurring in the community. In the United Kingdom there is a national syndromic surveillance system, operated jointly by the Health Protection Agency (HPA) and NHS Direct, which examines symptoms reported to NHS Direct. Data is analysed by the HPA and weekly bulletins are produced summarising NHS Direct call activity. Much of the published literature involves evaluating how good the surveillance system is at meeting its various aims (for example, Cooper *et al* [9], Baker *et al* [2], Doroshenko *et al.* [13]) and there are very few publications analysing NHS Direct data itself, for example using statistical models. Furthermore Smith *et al.* [48] suggest the need for work on space-time analysis of the data when they list integrating it into the surveillance

system as one of the future challenges.

One recent approach to modelling space-time count data is to use a descriptive Bayesian hierarchical model such as those of Mugglin *et al.* [36] and Knorr-Held and Richardson [27]. With complex Bayesian models such as these, the posterior is often difficult to evaluate and MCMC methods must be used. As noted in section 1.3, we can use the software `OpenBUGS` to implement this. Although `WinBUGS` (the Windows version) is probably best known, we found `LinBUGS` to be quicker for our model. The analysis of the output from `LinBUGS` can be performed through the `R` package `CODA`.

In this chapter we explore count data provided by the NHS Direct north east site using the space-time model of Mugglin *et al.* [36]. In sections 3.1 and 3.2 we describe the NHS data available to us and introduce the model. In section 3.3 we apply the model to the dataset using `LinBUGS` and in section 3.4 we interpret the results using the posterior distribution.

3.1 NHS Direct data

3.1.1 Data collection

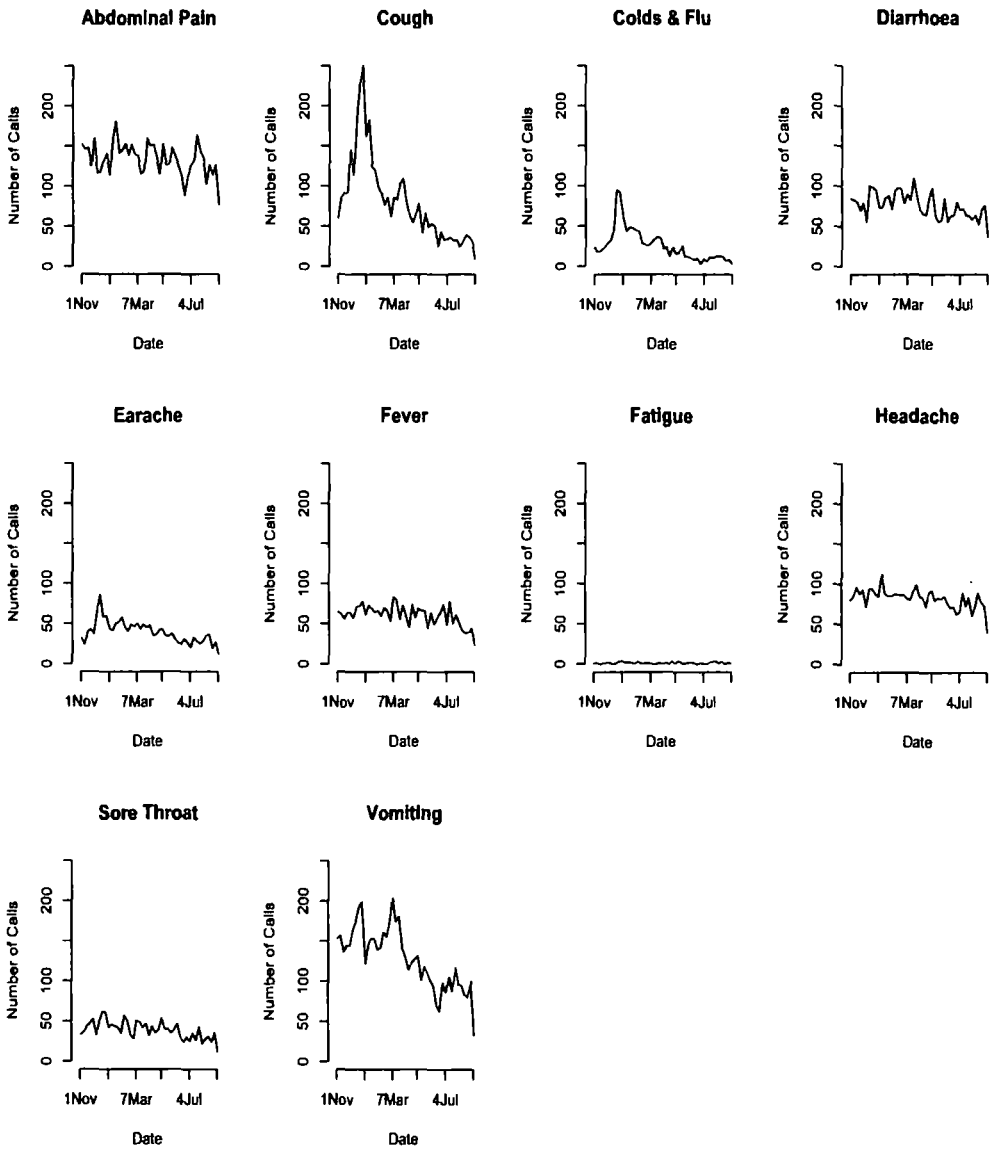
One of the roles of NHS Direct is to provide confidential health care advice. If you, or a member of your family, are feeling ill you can call the helpline and NHS Direct nurses will use their skills and experience, together with a comprehensive computer system, to advise you on the most appropriate course of action to take. The computer system involves algorithms consisting of a series of questions about the caller's symptoms. Computerised call records are held which include the patient's details, date and time of the call and which algorithms were used (this usually refers to the caller's main symptom). Information from the call records held at the NHS Direct north east site is what is available to us.

3.1.2 Exploring the data

The data set considered in this chapter covers a 45 week period from 1st November 2004 until 11th September 2005 and contains for each call: the date it was made, the Primary Care Trust (PCT) area of the caller, the age of the caller and the symptom they are calling about. The PCTs covered by the data set are Darlington, Derwentside, Durham and Chester Le Street, Durham Dales, Easington, Gateshead, North Tyneside, Newcastle, Northumberland, South Tyneside, Sedgefield and Sunderland, the locations of which are shown in Figure 3.2. The symptoms covered are Abdominal Pain, Cough, Colds and Flu, Diarrhoea, Earache, Fever, Fatigue, Headache, Sore Throat, Vomiting. To get an idea of what this data looks like, Figure 3.1 shows a plot of the number of calls received over the time period for each symptom.

We now decide to restrict attention to the symptom cough since it has an interesting temporal structure. The time series for this symptom shows an 'epidemic-like' pattern with the number of calls increasing from when we join the data set in November, peaking in the last week of December and then decreasing again following the peak. The number of calls reaches the highest peak (251 calls in one week) for the symptom cough and there are no weeks with no calls about a cough. Furthermore, all PCT areas are covered by the cough data set. The original data for this is given in the Appendix starting on page 164. Figure 3.2 shows us what this cough data set looks like spatially. It shows the total number of calls reported from each PCT during the whole 45 week period. We can see that there is a general pattern that the most southern areas have the least number of calls and increasing as we move towards the north. However, this representation of the data could be potentially misleading since it doesn't take into account the population of the areas. For example, Northumberland PCT is much larger than all of the others in area and also has the largest population so we would expect it to have a higher number of calls. When we model the data using the Bayesian model we adjust for the population (as described in section 3.3.3) so Figure 3.3 could be thought of as a more accurate spatial representation of the data. It shows the number of cough calls received from

Figure 3.1: Number of calls by symptom



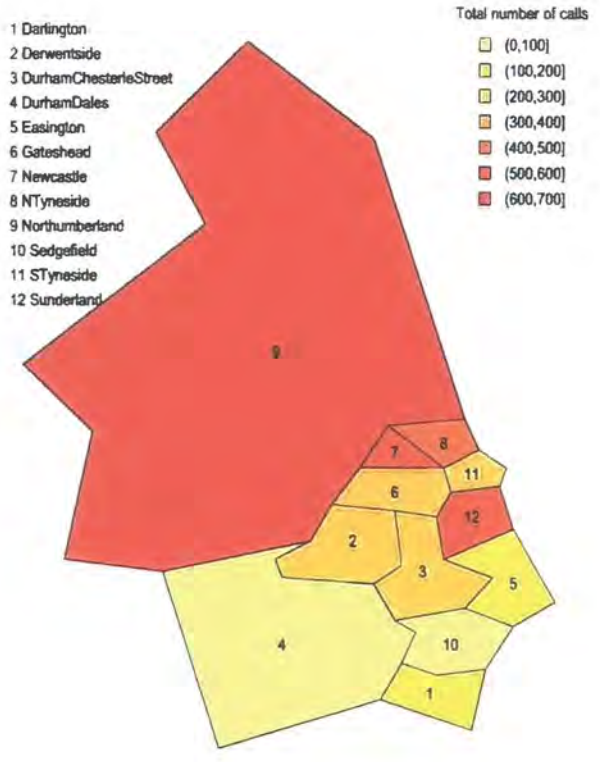


Figure 3.2: Spatial structure of the cough data

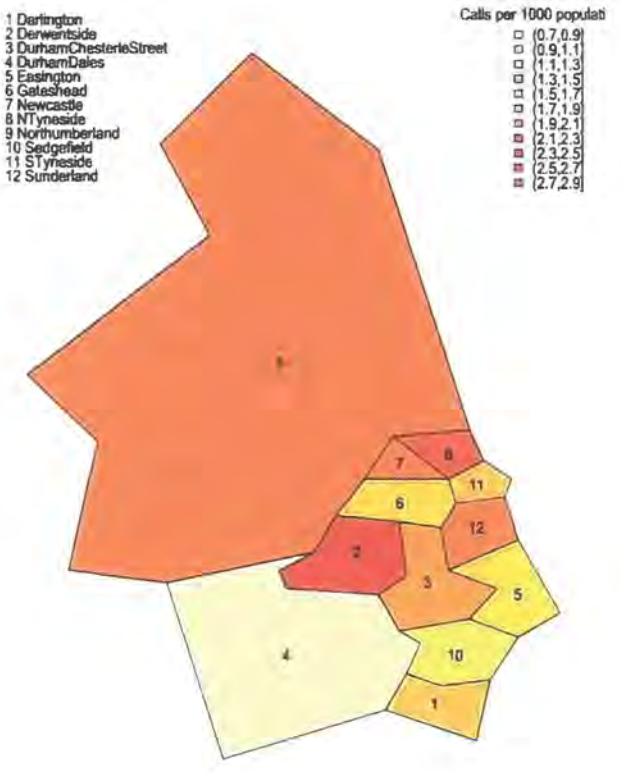


Figure 3.3: Spatial structure of the cough data scaled by population

each area per 1000 head of population so

$$scaled\ data = \frac{raw\ data}{PCT\ population} \times 1000.$$

Figure 3.3 shows that a north-south divide is not as clear now although the smallest rate of calls do appear to be in the south. Northumberland no longer shows the greatest number of calls, instead Derwentside does. This is because although Derwentside had one of the smallest number of calls in the raw dataset, it also had the smallest population available to make the calls. However, one potential disadvantage with the data set still stands in that the areas are geographically quite large. It could be that further levels of aggregation (such as postcode level) are necessary to draw any meaningful results about the spatial spread of the infection.

3.2 The model

Mugglin *et al.* [36] develop a hierarchical statistical model to be fitted to aggregated infectious disease data. The model attempts to capture both instantaneous spatial dependence as well as diffusion and growth in space and time.

Adopting this model, we let the number of calls Y_{it} from PCT i $(= 1, ..., 12)$ during week t $(= 1, ..., 45)$ be defined as

$$Y_{it} \sim Poisson(E_i e^{z_{it}})$$

where z_{it} is the logarithm of the relative risk and E_i is the number of calls expected to occur in PCT i in any one week under non-epidemic conditions. The way we calculate E_i by adjusting for age and population is described in section 3.3.3.

Covariates are used in the model to stratify the expected incidence but the main interest is in the space-time dynamics. If we had spatially or temporally varying explanatory variables they could be incorporated into the model by combining them into a vector $\mathbf{x}_{it}$ via

$$z_{it} = \mathbf{x}_{it}'\alpha + s_{it}$$

where α is a vector of regression coefficients. However, we choose not to include covariates and simply let $z_{it} = s_{it}$.

For $t = 1$ we define $\boldsymbol{s}_1 \sim MVN(\mathbf{0}, \zeta^2\Sigma)$ where $\zeta^2 > 1$ is chosen to reflect additional uncertainty about $\boldsymbol{s}_1$. For $t = 2, ..., 45$ we use the multivariate Gaussian autoregressive process to define

$$\boldsymbol{s}_t = H\boldsymbol{s}_{t-1} + \boldsymbol{\varepsilon}_t \tag{3.1}$$

where H is an 12×12 autoregressive coefficient matrix and $\boldsymbol{\varepsilon}_t$ is the epidemic forcing term which is assumed to be a realisation from a Gaussian Markov random field. Specifically,

$$\boldsymbol{\varepsilon}_t \sim MVN(\beta_{\rho(t)}\mathbf{1}, \Sigma) \tag{3.2}$$

where Σ is the variance-covariance matrix and $\rho(t) = 0, 1$ or 2 indicating stage of

disease at time t as given by

$$\beta_{\rho(t)} = \begin{cases} \beta_0 & \text{if } t < t_0, \text{ for stability} \\ \beta_1 & \text{if } t_0 \leq t < t_1, \text{ for growth} \\ \beta_2 & \text{if } t_1 \leq t < t_2, \text{ for intermediate decline} \\ \beta_0 & \text{if } t \geq t_2, \text{ for final decline to stability} \end{cases}$$

Σ is from the conditional autoregressive (CAR) class of models and is defined by

$$\Sigma = \sigma^2(I - \phi C)^{-1}M \tag{3.3}$$

M is a diagonal matrix with entries E_i^{-1} on the diagonal and $c_{ij} = (E_j/E_i)^{1/2}$ if site j is a neighbour of site i and 0 otherwise. σ^2 is the variance related to spatial association and $\phi \in (\phi_{min}, \phi_{max})$ is the spatial dependence parameter where ϕ_{min} and ϕ_{max} are determined from the eigenvalues of C such that $M^{-1}(I - \phi C)$ is positive definite. Spatial dependence is also included in the structure of H which is parametrised by η_0, η_1 and η_2 as follows

$$h_{ij} = \begin{cases} \eta_0 & \text{if } j = i \\ \eta_1 & \text{if } j \in N_i, \text{ that is, j is neighbour of i} \\ \eta_2 & \text{if } j \in N_i^{(2)}, \text{ that is, j is second-order neighbour of i} \\ 0 & \text{otherwise} \end{cases}$$

η_0 can be interpreted as a global measure of how much any site is affected by itself at one previous time lag while η_1 and η_2 are global measures of the impact of the first and second order neighbours, respectively, at one previous time lag. Instead of assigning a prior distribution directly to the η_ℓ ($\ell = 0, 1, 2$) they are transformed using $\theta_\ell = log[(1 + \eta_\ell)/(1 - \eta_\ell)]$ and a prior is assigned to the θ_ℓ. The model is completed by specifying the following priors

$$\beta_\ell \sim Normal(\mu_{\beta_\ell}, \tau_{\beta_\ell}^{-1}), \quad \ell = 0, 1, 2 \tag{3.4}$$

$$\theta_\ell \sim Normal(\mu_{\theta_\ell}, \tau_{\theta_\ell}^{-1}), \quad \ell = 0, 1, 2 \tag{3.5}$$

$$\sigma^2 \sim InverseGamma(a, b) \tag{3.6}$$

$$\phi \sim Uniform(\phi_{min}, \phi_{max}) \tag{3.7}$$

3.3 Using LinBUGS

To run an MCMC simulation in `LinBUGS`, four files are needed. One containing the actual script commands, one with the `BUGS` language representation of the model, one containing the data and one containing the initial values. `LinBUGS` is an 'expert system' which attempts to use the most appropriate sampling scheme for each parameter. It has a hierarchy of methods and a particular sampling scheme is used if no previous method in the hierarchy is appropriate. It starts with direct sampling using standard algorithms, then if that is not appropriate it uses derivative-free adaptive rejection sampling, then slice sampling, and so on. Further details of the hierarchy can be found in the `WinBUGS` manual [1]. When a Metropolis MCMC algorithm is used, it is based on a symmetric normal proposal distribution whose standard deviation is tuned over the first 4000 iterations in order to get an acceptance rate of between 20% and 40%.

3.3.1 The script file

The first commands needed in the script file are

```
modelCheck("model.txt")
modelData("data.txt")
modelCompile()
modelInits("inits.txt",1)
```

which first checks that the model description fully defines a probability model and reports any syntax errors. Next, the data is loaded and the model is compiled. This sets up the internal data structures needed to carry out the MCMC sampling and chooses the specific MCMC updating algorithms to be used for this particular model. The model is also checked for completeness and consistency with the data. Once the model has been successfully compiled, the MCMC sampler must be given some initial values for each stochastic node and the number of chains to simulate is chosen (one in this case). Checks on the initial values are then carried out to ensure

[1] http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/manual14.pdf

they are of the correct form and are consistent with any previously loaded data. Any syntax errors or inconsistencies are displayed. The next commands we use are

```
modelUpdate(1000,1)
samplesSet(beta0)
modelUpdate(10000,130)
samplesCoda(beta0, "beta0")
```

which first performs 1000 burn in updates which are to be discarded and then sets the monitors to start recording the values sampled for the parameter `beta0`. The simulation is then run again, this time thinning every 130 iterations, until a total of 10000 values are recorded. The final command is asking `LinBUGS` to produce the MCMC output for `beta0` in `CODA` format which allows it to be read into `R`.

3.3.2 Specifying the model

The model can be specified using the text-based BUGS language which allows it to be expressed concisely. The $\sim$ symbol denotes stochastic relationships and $<-$ denotes deterministic ones. The model specification also allows arrays, loops, nested indexing and has a range of integrated functions. We also make use of the add-on module `GeoBUGS` which allows us to create and manipulate matrices necessary for conditional autoregressive models. In particular, we rewrite equations (3.1) and (3.2) as

$$\boldsymbol{s}_t \sim MVN(\beta_{\rho(t)}\mathbf{1} + H\boldsymbol{s}_{t-1}, \Sigma)$$

and use the `car.proper` function to specify it. The Mugglin *et al.* [36] model can be represented in this `BUGS` language as follows:

```
model {
    for (i in 1:I) {
        for (t in 1:T) {
            Y[t,i] ~ dpois(lambda[t,i])
            lambda[t,i] <- E[i]*exp(s[t,i])
        }
        for (t in 2:T) {
            hs[t,i]<-inprod(h[,i],s[t-1,])
```

```
        }
        for (t in 2:(t0-1)) {
            mu.s0[t,i]<-hs[t,i]+beta0
        }
        for (t in t0:(t1-1)) {
            mu.s0[t,i]<-hs[t,i]+beta1
        }
        for (t in t1:(t2-1)) {
            mu.s0[t,i]<-hs[t,i]+beta2
        }
        for (t in t2:T) {
            mu.s0[t,i]<-hs[t,i]+beta0
        }
        for (j in 1:I) {
            h[i,j]<-(equals(H[i,j],9)*eta0)
                    +(equals(H[i,j],1)*eta1)
                    +(equals(H[i,j],2)*eta2)
        }
        mu.s[i]<-mu.t1
    }
    for (t in 2:T) {
        s[t,1:I] ~ car.proper(mu.s0[t,],C[],adj[],num[],M[],
                    invsigma.sqd,phi)
    }
    s[1,1:I] ~ car.proper(mu.s[],C[],adj[],num[],M[],tau.s,phi)
    phi ~ dunif(phi.min,phi.max)
    phi.min<-min.bound(C[],adj[],num[],M[])
    phi.max<-max.bound(C[],adj[],num[],M[])
    invsigma.sqd ~ dgamma(a,b)
    sigma.sqd<-1/invsigma.sqd
    tau.s<-invsigma.sqd*(1/xi.sqd)
```

```
    eta0<-(exp(theta0)-1)/(exp(theta0)+1)
    eta1<-(exp(theta1)-1)/(exp(theta1)+1)
    eta2 <-(exp(theta2)-1)/(exp(theta2)+1)
    theta0 ~ dnorm(mu.theta0,tau.theta0)
    theta1 ~ dnorm(mu.theta1,tau.theta1)
    theta2 ~ dnorm(mu.theta2,tau.theta2)
    beta0 ~ dnorm(mu.beta0,tau.beta0)
    beta1 ~ dnorm(mu.beta1,tau.beta1)
    beta2 ~ dnorm(mu.beta2,tau.beta2)
}
```

3.3.3 Specifying the data

The data file is where we enter the count data Y_{it} as well as the value of hyper-parameters and other constants in the model. This file can be represented using R object notation and values are given in a single structure headed by the key-word 'list' as follows

```
list(   T = 45, I=12,
        E=c(10.698476, 7.848950,..., 3.960951, 3.601294),
        mu.theta0=0, tau.theta0=0.25, mu.theta1=0, tau.theta1=0.25,
        mu.theta2=0, tau.theta2=0.25, mu.beta0=0, tau.beta0=0.25,
        mu.beta1=0, tau.beta1=0.25, mu.beta2=0, tau.beta2=0.25,
        a=0.25, b=2.5, mu.t1=0, xi.sqd=4, t0=4, t1=10, t2=15,
        M=c(0.093471, ..., 0.277677), C= c(0.770000, ..., 1.048746),
        num=c(2,4,...,4,4), adj=c(4,10,...,6,11),
        H=structure(.Data=c(9, 2, 2, 1,..., 2, 1, 9),
                    .Dim = c(12, 12)),
        Y=structure(.Data=c(3, 3, 9, 0, ..., 2, 0, 0, 1),
                    .Dim = c(45, 12)),
    )
```

We now explain our choice of data values a bit further. The first details we need to include are the total number of PCTs and weeks which we do via $I = 12$ and $T = 45$.

The expected number of calls E_i for each PCT in any one week under 'non-epidemic' conditions were calculated using the method of Mugglin *et al.* [36] who adjust for demographic effects. We divide the calls into 19 strata based on age (0-4, 5-9, ..., 85-89, 90+) and obtained the population for each PCT and each age bracket from the 2001 census. We then calculated

$$E_i = \sum_{k=1}^{19} R_{ik} q_k$$

where R_{ik} is the population in PCT i and stratum k and q_k is the proportion in stratum k expected to become a case estimated by

$$\widehat{q_k} = \frac{\sum_t \sum_{i=1}^{12} y_{ikt}}{\sum_t \sum_{i=1}^{12} R_{ikt}}$$

where y_{ikt} is the observed number of calls in PCT i in the kth stratum during week t. We assume that the population doesn't change over the time period so R_{ikt} is constant. The resulting expected number of calls per week range from 3.44 for Newcastle PCT to 12.23 for Gateshead PCT.

Next we choose the same prior values as used in [36] namely

$$\begin{aligned} \beta_\ell &\sim Normal(0, 4), \quad \ell = 0, 1, 2 \\ \theta_\ell &\sim Normal(0, 4), \quad \ell = 0, 1, 2 \\ \sigma^2 &\sim InverseGamma(0.25, 0.4) \end{aligned}$$

and include these via the '`mu.theta0`=0, `tau.theta0`=0.25, ..., `a`=0.25, `b`=2.5' commands.

The `mu.t1` and `xi.sqd` just represent the mean and ζ^2 values in the specification $s_1 \sim MVN(\mathbf{0}, \zeta^2\Sigma)$ which were chosen to be the same as those used in [36].

The values for t_0, t_1 and t_2 were obtained from a more detailed version of the time series plot shown in Figure 3.1. We find that the 'epidemic' is stable until week 4, growing until week 10 and undergoing intermediate decline until week 15 when it begins a final decline to stability.

The car.proper command requires data to be added about the between-area covariance matrix Σ. Recall from equation (3.3) that Σ can be written as $\sigma^2(I-\phi C)^{-1}M$ where M is a diagonal matrix with entries E_i^{-1} on the diagonal and $c_{ij} = (E_j/E_i)^{1/2}$ if site j is a neighbour of site i and 0 otherwise. We therefore need to include C and M in the data file as well as information about the neighbourhood structure in the form of two vectors 'num' and 'adj' (num gives the number of neighbours each area has and adj lists the ID numbers of each adjacent area). C takes the form of a vector the same length as adj giving the weights associated with each pair of areas and M is just the vector $(\frac{1}{E_1}, \ldots, \frac{1}{E_{12}})$.

Another piece of information we need to include is the value of the autoregressive coefficient matrix H. Recall from section 3.2 that each h_{ij} takes one of the values η_0, η_1, η_2 or 0 depending on the neighbourhood structure. This is included in the data file via the 'structure' function where 9 corresponds to a value of η_0, 1 to a value of η_1 and 2 to a value of η_2. This reads the string of numbers specified as '.Data' into a 12×12 dimensional array.

Finally, the count data is included in the file again using the structure function. This time it reads the values specified in '.Data' into a 12×45 dimensional array.

3.3.4 Initial values

It is necessary to choose some initial values for each stochastic node. Although BUGS has an option to generate you some initial values, choosing your own gives you more control. BUGS generates values using the prior distributions but when the priors are vague, initial values generated may be at the extremes of the distributions, producing an error message when the model is run. Different sets of initial values were tried and we decide to use the following set

$$\theta_0 = 0.01, \theta_1 = 0.01, \theta_2 = 0.01, \beta_0 = 0.01, \beta_1 = 0.01, \beta_2 = 0.01,$$
$$\sigma^{-2} = 5, \phi = 0.1, s_{it} = 0.01 \text{ for all i and t}$$

which are chosen to be close to the centre of their prior distributions. We know these initial values are not inappropriate as convergence is relatively fast. Convergence refers to the idea that the MCMC technique used will eventually reach a stationary

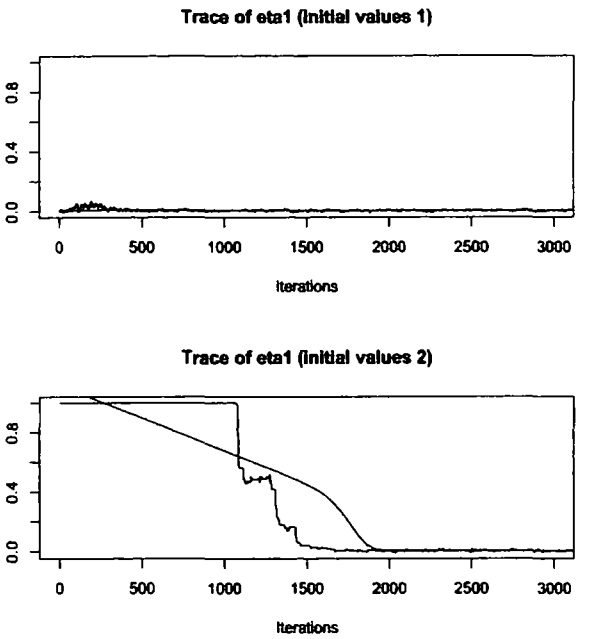


Figure 3.4: Traceplots resulting when two different sets of initial values are used

distribution. A traceplot is a diagnostic tool which plots the parameter value at time t against the iteration number. If the model has converged then the traceplot will snake around close to the mode of the distribution whereas a sign of non-convergence would be some sort of trending pattern. Figure 3.4 shows traceplots for one of the model variables, namely η_1. The top plot results when the above initial values are used and the bottom one results when a changed set of initial values are used. This second set changes $\theta_0, \theta_1, \theta_2, \beta_0, \beta_1$ and β_2 from 0.01 to 100. We can see that convergence is obtained more quickly for the original set of initial values.

3.3.5 Output

Once the model is successfully compiled and initialised we do a short pilot run of the chain in order to check convergence and decide how many updates need discarding to allow for burn-in. This is because we want to be sure that the chain has reached the stationary distribution and has 'forgotten' the starting values before we start any summarising of the target distribution. After obtaining MCMC

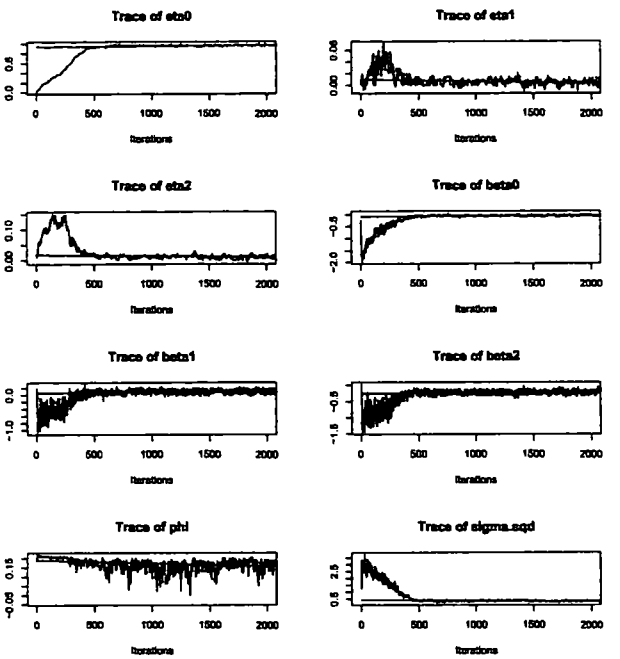


Figure 3.5: Traceplots used as an informal check of convergence

output from LinBUGS for each monitored variable, we then read it into R using the `read.openbugs()` command in the `CODA` package. Next we do an informal check of convergence using traceplots as described in section 3.3.4. Figure 3.5 shows examples of traceplots for some of the variables of interest and we can see that convergence appears to occur by about 1000 updates for all of the variables shown. A more formal way of determining the number of initial iterations to discard is to use a diagnostic such as that of Raftery and Lewis [38]. The `raftery.diag()` function is included in the `CODA` package and provides a way for us to easily calculate the suggested burn-in for each of the variables of interest using a short pilot run of a Markov chain. The diagnostic is based on a criterion of accuracy of estimation of the quantile q. The number of iterations required to estimate the quantile q to within an accuracy of $\pm r$ with probability p is calculated. In the `CODA` package the default values $q = 0.025$, $r = 0.005$ and $p = 0.95$ are used and separate calculations are performed for each variable within each chain. Using this diagnostic, the variable which we find to need the highest number of iterations discarded is η_0 which needs 1897. We therefore choose to discard 2000 initial iterations as burn-in.

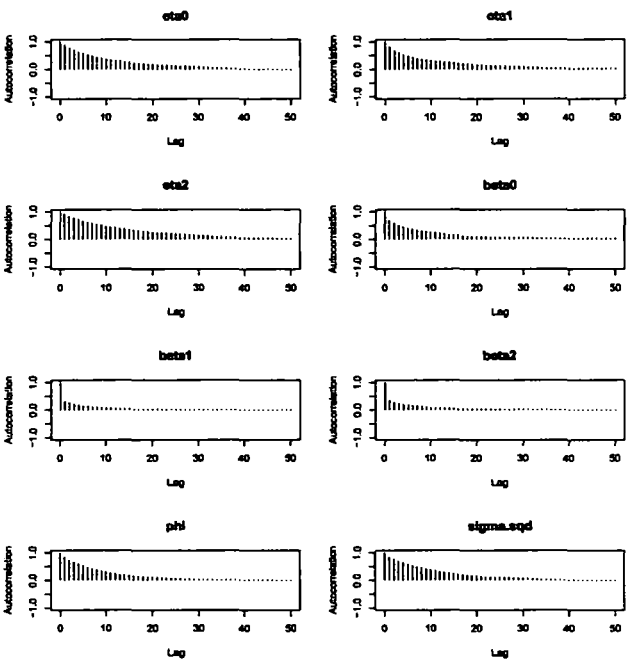


Figure 3.6: Autocorrelation functions before thinning

After allowing for burn-in, we set the monitors to start recording the values sampled for each parameter. We then do another short run to look at the autocorrelation for each variable. Autocorrelation refers to a pattern in the chain where sequential draws of a parameter from the conditional distribution are correlated. High autocorrelation would result in the Gibbs sampler being slow to explore the entire posterior distribution. This would mean that larger sample sizes would be necessary to make credible Bayesian inferences therefore increasing computational effort. Figure 3.6 shows plots of the autocorrelation functions for all variables of interest apart from the relative risks. (Note that autocorrelation functions for each s_{it} were checked separately and found to be lower than those shown in Figure 3.6). Typically plots of autocorrelation functions will decline with an increasing number of lags but ideally we would like it to be near zero for all lags. We can see from these plots that the autocorrelations for some variables still seem to be relatively high out to a lag of around 30 to 40. Autocorrelation can be addressed by 'thinning' the Markov chain. Thinning an MCMC chain means that not all samples are stored but are recorded periodically at a rate that can be specified. In our case we choose

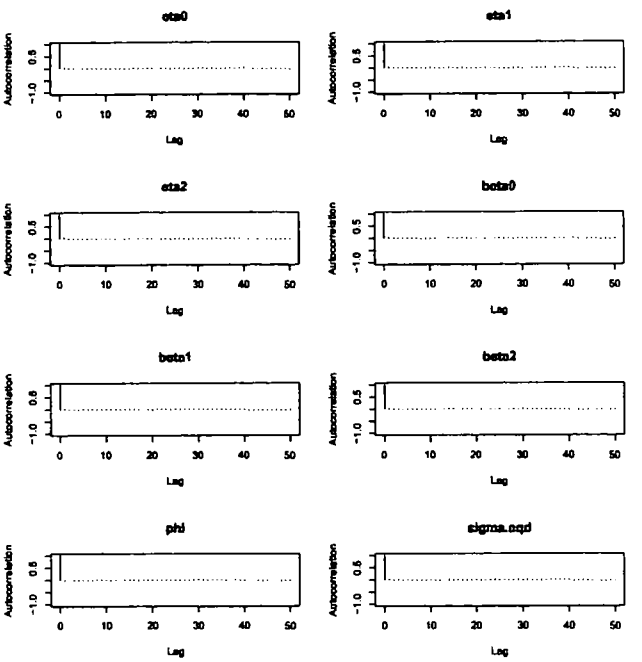


Figure 3.7: Autocorrelation functions after thinning

to run the sampler again, this time recording every 50th sample only until we have a sample size of 10000. We then redo the autocorrelation plots which are shown in Figure 3.7 and they show us that correlation for each parameter has decreased to an acceptable level. It then remains for us to interpret the results using the posterior distribution.

3.4 Interpreting the results

3.4.1 Variables of interest

Recall from section 3.2 that we have the following variables of interest:

- η_0 is the parameter which shows the lag-1 influence of an area on itself which we would expect to be > 0. η_1 and η_2 show how much an area is affected by its first-order and second-order neighbour's value from the previous week, respectively. η_1 and η_2 will give us an idea of whether there is any spatial structure over time. A value of zero will indicate that there is no spatial

structure.

- β_0, β_1 and β_2 represent the size of the epidemic in different stages: stability, growth and decline respectively. We would therefore expect the posterior densities to show that $\beta_1 > \beta_0 > \beta_2$.
- ϕ is the spatial dependence parameter which can be interpreted as the partial correlation squared between two areas within any week, i.e. $corr^2\{\epsilon_{it}, \epsilon_{jt} \mid \text{rest of } \epsilon_t\}$. We can therefore get an idea of how spatially dependent any two areas are within time. If $\phi = 0$ then this would indicate no spatial dependence.
- σ^2 controls the overall variability of the epidemic forcing term ϵ_t.
- s_{it} is the logarithm of the relative risk associated with area i at time t.

3.4.2 Posterior densities

We first note that the posterior credible intervals are much tighter than those of the priors, for example see Table 3.1 which shows the prior and posterior summaries for the β_i, η_i, ϕ and σ parameters. This means that there is substantial information in the dataset to infer about the parameters.

	Prior			Posterior		
	0.025	0.500	0.975	0.025	0.500	0.975
β_0	-3.920	0.000	3.920	-0.114	-0.048	0.012
β_1	-3.920	0.000	3.920	0.003	0.108	0.207
β_2	-3.920	0.000	3.920	-0.335	-0.219	-0.104
η_0	-0.961	0.000	0.961	0.939	0.967	0.991
η_1	-0.961	0.000	0.961	-0.004	0.005	0.014
η_2	-0.961	0.000	0.961	-0.001	0.011	0.023
ϕ	-0.325	-0.075	0.163	0.064	0.160	0.198
σ^2	1.456	57.243	9.5×10^6	0.235	0.325	0.446

Table 3.1: Prior and posterior quantiles

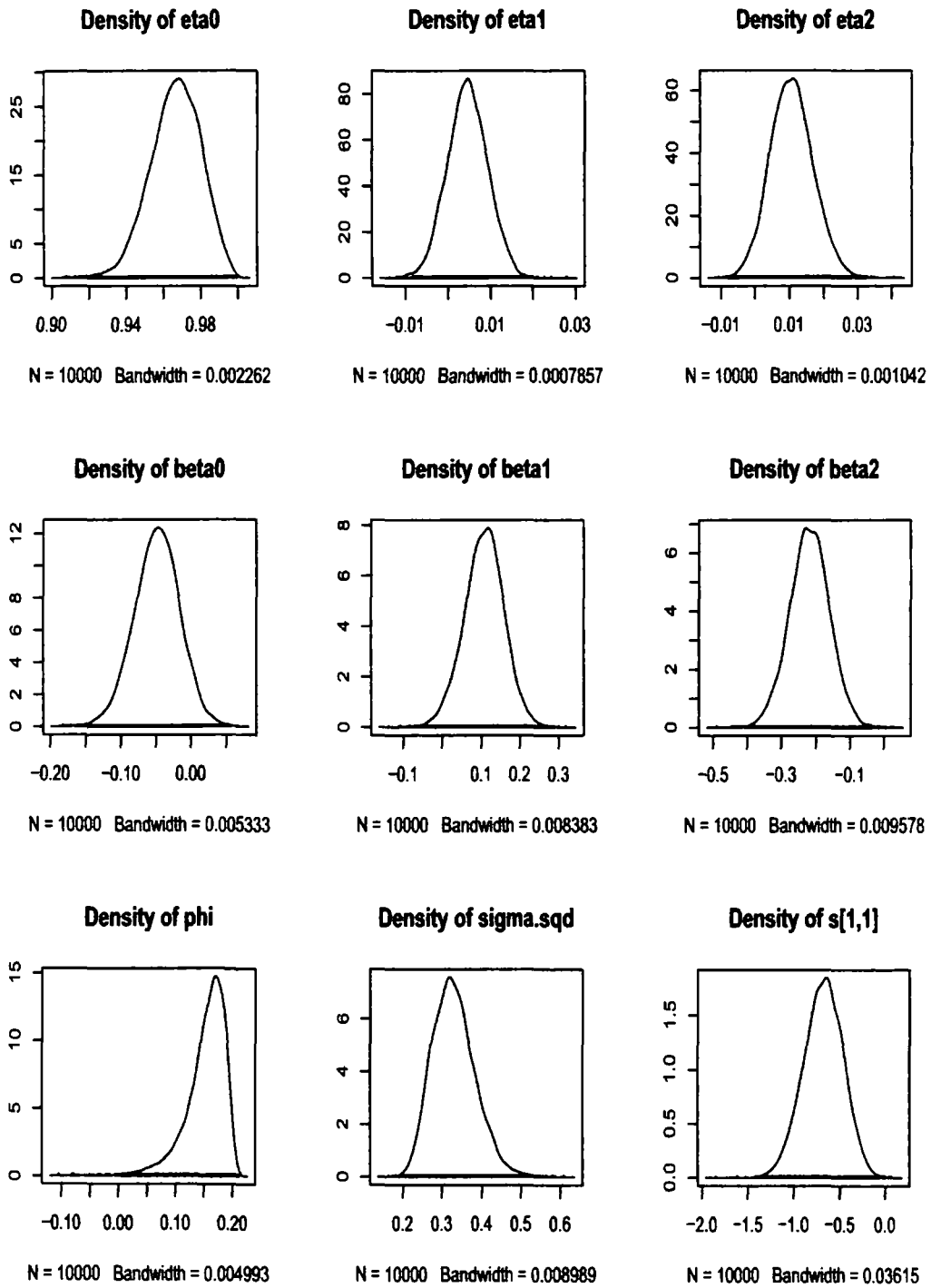


Figure 3.8: Posterior densities

Figure 3.8 shows plots of the posterior densities for the variables of interest although only one of the s_{it} are shown. We were particularly interested in whether there is any spatial structure in the spread of infection in the North East area. We can see that there does seem to be some degree of spatial structure: η_0 is shown to be definitely positive and η_1 and η_2 also seem likely to be positive, although they are small in relation to η_0. It is strange that η_2 is larger than η_1 since they correspond to second order neighbours and first order respectively. A possible explanation is the small number of regions we have available for our data set. For example, if we look at Figure 3.3 on page 42 we can see that the layout of the 12 regions means that practically every region is either a first order or second order neighbour of every other region. However, the positive value of the ηs do suggest that there is some structure present over time. Furthermore, ϕ has a median of 0.16 and 95% Bayes credible interval of [0.064,0.198] so this is positive indicating there is some spatial correlation within time.

3.4.3 Representations of the posterior relative risk

The posterior relative risks, $e^{\theta_{it}}$, represent smoothed values of the raw standardised morbidity ratios, y_{it}/E_i, and therefore give us a smoothed picture of what is going on with the 'epidemic'. Figure 3.9 shows the average logarithm of these posterior relative risks over 5 week periods. One thing that we notice from each of the pictures is that the more southern PCTs tend to have a lower relative risk than the northern ones, this is consistent with the observations we made about the raw count data in section 3.1.

Figure 3.10 shows the pattern of the s_{it} over time broken down to show each PCT separately. The question arises as to whether any of these time series differ from each other and if so, is there any spatial structure in the differences? From first glance it appears that some are higher in general, for example Newcastle, North Tyneside, Northumberland and Sunderland. It is also clear that some start from lower and increase steadily to a peak, for example North Tyneside, whereas others start with a dip before increasing by a smaller amount to the peak, for example Sedgefield.

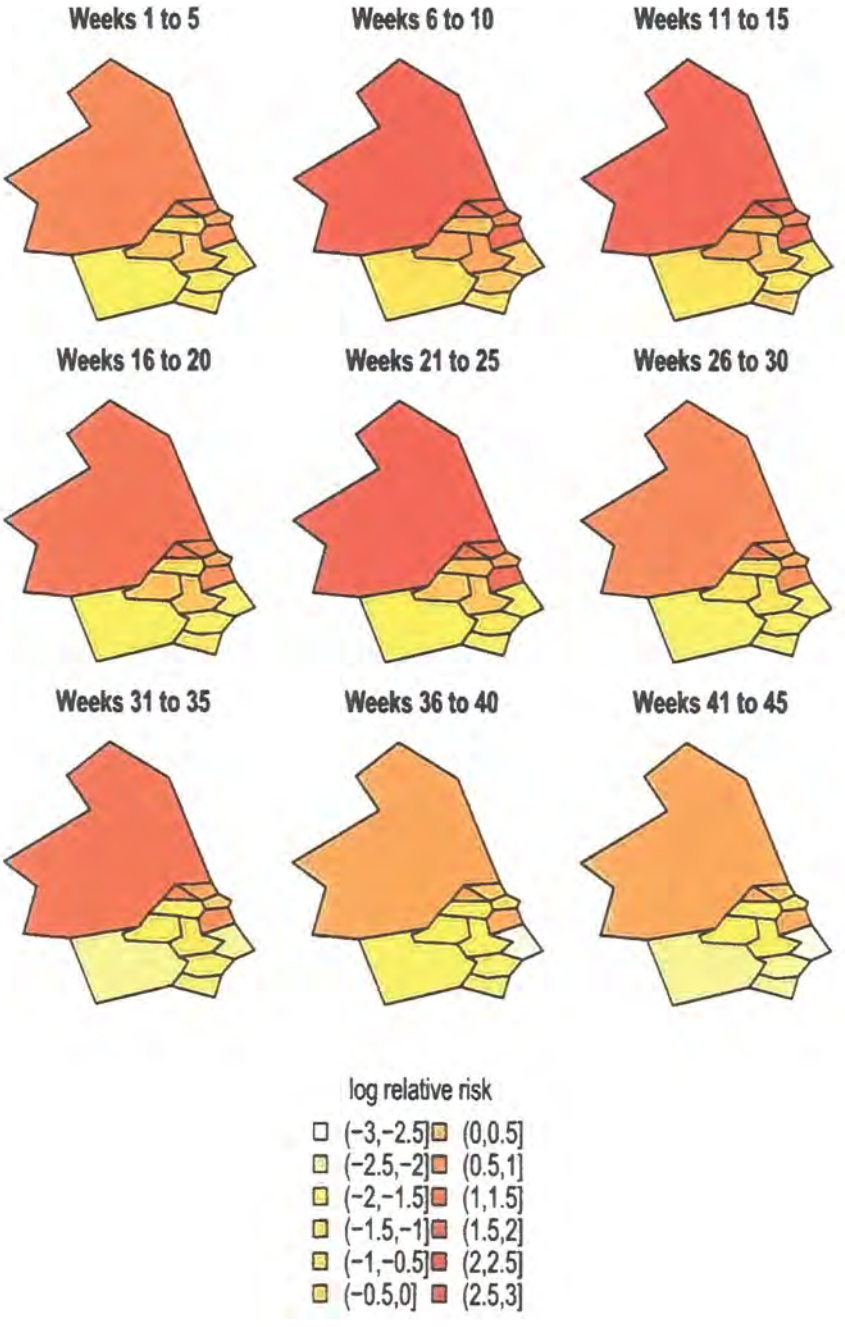


Figure 3.9: Average s_{it} for grouped time periods

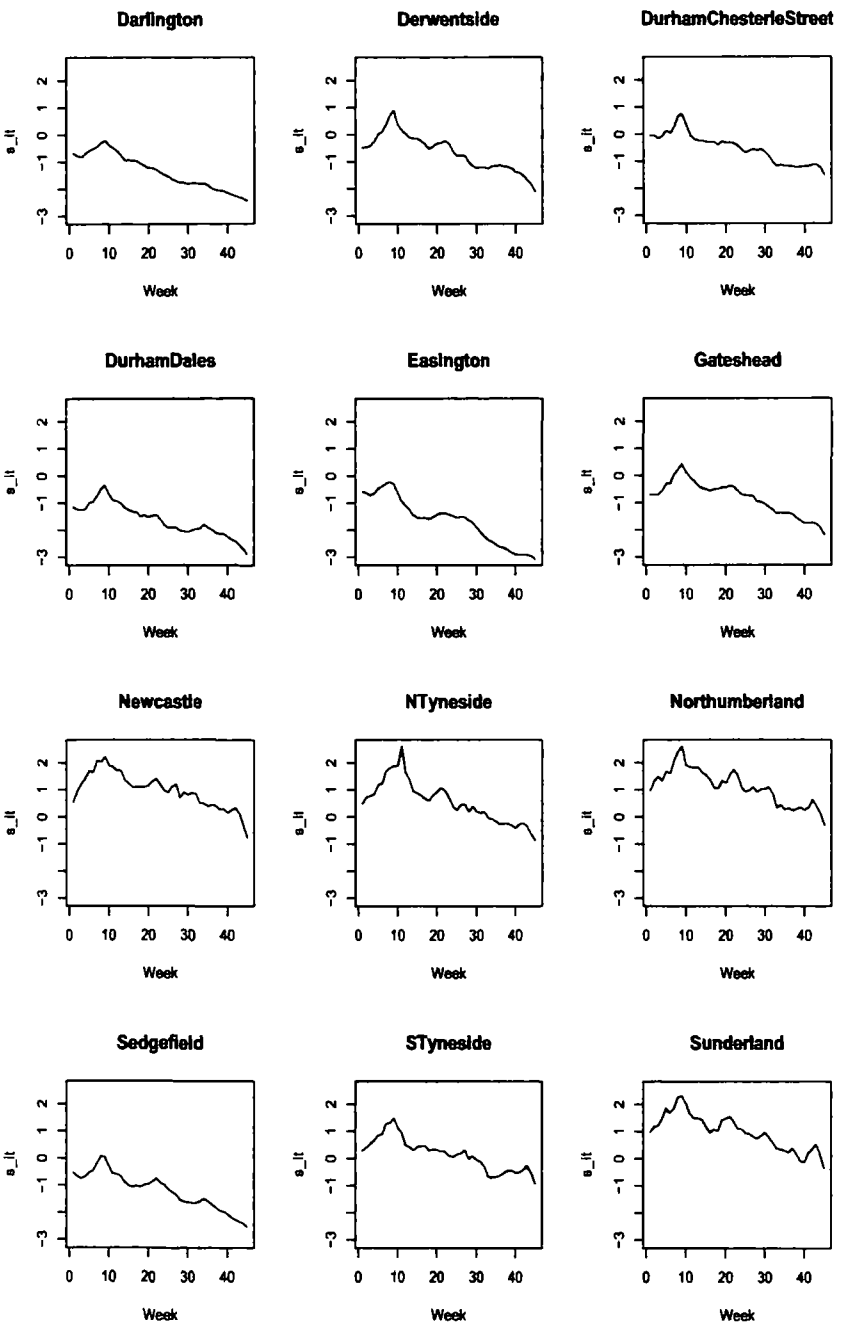


Figure 3.10: s_{it} over time for each PCT

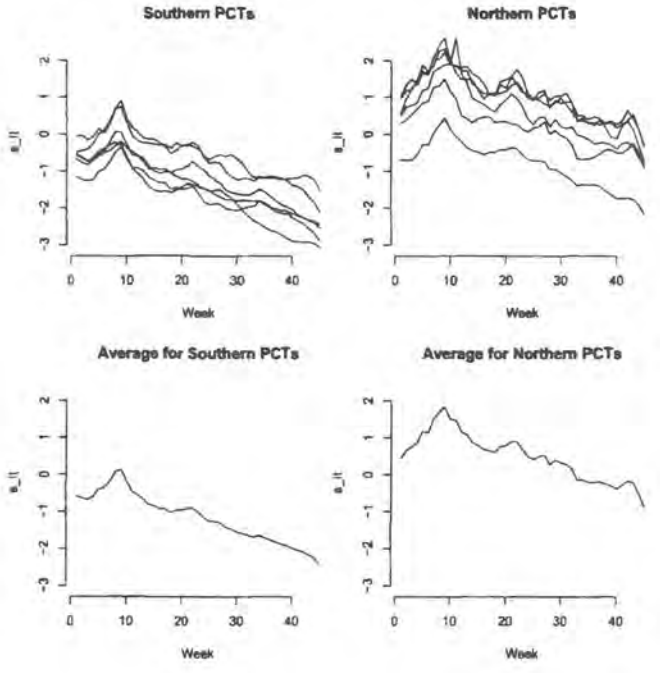


Figure 3.11: s_{it} over time for PCTs grouped by northern or southern

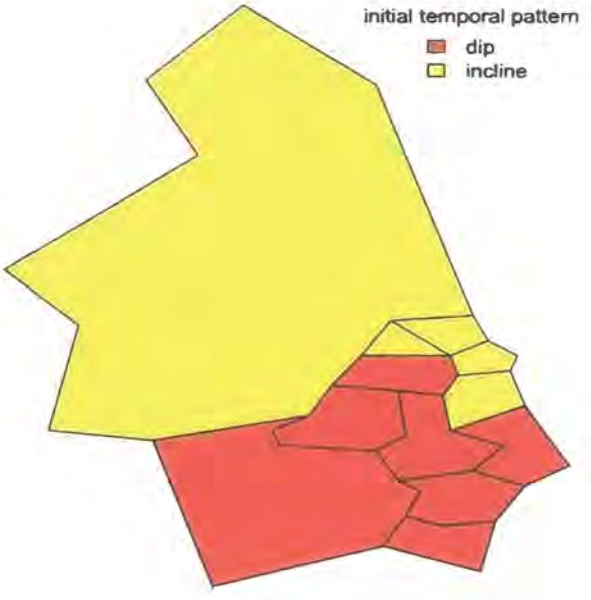


Figure 3.12: Spatial structure of the initial behaviour of the temporal plots

It appears that these differences could be between those PCTs in the north and those in the south of our study region. Figure 3.11 shows the temporal patterns grouped separately for northern and southern areas as well as their averages. The six most northern PCTs can be seen to have an average s_{it} which is generally higher than that of the southern ones and also starts with a steady incline rather than a dip. In order to picture this, Figure 3.12 separates the PCTs into red or yellow depending on whether the initial behaviour of the relative risk shows a dip or an incline as defined by eye using 3.10.

3.4.4 Is the NHS Direct data reliable?

There is some question as to whether NHS Direct data is useful for telling us about the spread of infection in the north east area. Smith *et al.* [48] point out one disadvantage which is that the NHS call rate is low when compared to the consultation rate for GPs so it only captures a small proportion of illness reported to primary care. It has also been suggested by Cooper *et al.* [8] that factors such as deprivation and age are likely to affect the uptake of the service meaning that it may not reflect the health of all sections of the population. There is also the fact that during peak times or times of staff training, calls are rerouted to other call centres so there may be calls made from patients in the 12 PCTs that wouldn't have been included in the data we received from the NHS Direct north east site. Furthermore, it could be said that the PCT area is too large to be able to draw any meaningful conclusions about spatial structure and aggregations such as postcode would have been better. However, this was not possible due to reasons of confidentiality.

3.5 Model Assessment

A range of methods exist to assess our model. One such technique is to use a posterior predictive distribution and examine the extent to which the replicated values from this distribution match the original data. One could also perform a residual analysis and look for outliers but this can be quite complex to compute for our type of model. We could also check overall model fit by fitting models of

varying complexity and comparing them. In our case we could make the model simpler by removing the spatial term altogether or more complex by including a more complicated neighbourhood structure. The classical way to compare models is to use Bayes factors but these can be difficult to calculate for our type of model and in recent years the deviance information criterion (DIC) has become more popular since it can be computed easily from MCMC output. However it's properties for more complex models are not as well understood. A further method of assessment is to do a prior sensitivity analysis which we have considered in detail for this model in chapter 5.

Chapter 4

Improving the efficiency of MCMC

In this chapter we consider ways of improving the efficiency of MCMC for Poisson regression models such as that of Mugglin *et al.* [36] described in chapter 3. By efficiency we mean the speed and ease with which a simulated sample can be obtained.

4.1 Motivation

Although the NHS Direct data set modelled in chapter 3 was relatively quick to run in `LinBUGS`, we found this not to be the case for a larger data set. Data with 95 spatial areas and 52 time points took approximately 100 hours to get a sample of size 1 using the same set up. The problem may be due to a slow sampling scheme being used for some of the parameters. `LinBUGS` is an 'expert system' that attempts to use the most appropriate sampling scheme for each parameter. We know from section 1.2 that using Gibbs sampling can be quite fast but requires that each conditional distribution has a recognised distributional form. However when we look more closely at the set up of the Poisson regression model, we see that the conditional distribution of each parameter (or block of parameters) does not have a standard form and would therefore require `LinBUGS` to choose an alternative sampling scheme such as Metropolis-Hastings. In particular, for the model considered in chapter 3, we discover that $p(\mathbf{s} \mid \text{rest})$ is not of standard form. It involves generating from a multivariate Normal density multiplied by product of Poisson densities where the

rate involves an exponential. More formally,

$$\begin{aligned} p(\mathbf{s} \mid \text{rest}) &\propto p(\mathbf{s} \mid \beta_0, \beta_1, \beta_2, \sigma^2, \phi, \theta_0, \theta_1, \theta_2) \prod_{i=1}^{I} \prod_{t=1}^{T} p(y_{it} \mid s_{it}) \\ \text{where } Y_{it} &\sim Poisson(E_i e^{s_{it}}) \\ \boldsymbol{s} &\sim MVN(\boldsymbol{\mu}, \Sigma) \end{aligned} \tag{4.1}$$

Although Metropolis-Hastings can work well, we mentioned in section 1.2 that it can be difficult to propose good candidate values for complex models such as this and that it requires tuning. It may therefore improve the efficiency if we could make this non-standard conditional distribution into a standard one and then use block Gibbs sampling. Our overall goal in this chapter is to make (4.1) take the form of a multivariate Normal distribution by augmentation in such a way that the basic structure of its precision matrix remains the same even when the augmenting variables and other parameters change. We also want it to be easy to sample the augmenting variables given s. Furthermore, for the sampling scheme to be efficient we need to ensure that when the other parameters change, we can obtain a sample of s using only a relatively small number of alternating s and augmenting variable block Gibbs steps. We begin by simplifying our problem to a univariate case.

4.1.1 Simplified example

Suppose that we want to use Gibbs sampling to generate from

$$p(x \mid y) \propto exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu)^2 \Big\} e^{xy} e^{-e^x} \tag{4.2}$$

where

$$\begin{aligned} X &\sim N(\mu, \sigma^2) \\ Y \mid X &\sim Poisson(e^X) \end{aligned}$$

for known μ, σ and y. Our problem is to find a way to make (4.2) take the form of a Normal distribution by augmentation.

4.2 Current research in this area

As we mentioned in section 1.2.1, a similar problem has been addressed by Damien *et al.* [11] who introduce auxiliary variables to enable the sampling of non-standard densities using the Gibbs sampler in the context of Bayesian non-conjugate and hierarchical models. Adopting their approach in our case would result in a truncated Normal distribution. While this would provide the solution for the univariate example, it would not be useful for the multivariate case since generating from a truncated multivariate Normal distribution is not trivial. While it could be achieved using single variable Gibbs sampling as in Rodriguez-Yam *et al* [42], our aim is to use block Gibbs sampling with large blocks in order to improve the efficiency. In section 4.3 we generalise the idea presented in [11] and in section 4.4 we look at possible ways of adapting it to work for our example. However, we are unable to make the adaptations work in such a way that improves the efficiency of the sampler.

An alternative approach for tackling such problems is known as auxiliary mixture sampling and has been an active research area over recent years. It was introduced in Shephard [47] for stochastic volatility models and has been applied in this context by Kim *et al.* [24] and Chib *et al.* [5], among others. Recently, the method has been extended to more general hierarchical models for non-Gaussian data. Frühwirth-Schnatter and Wagner [17] present the method for Poisson regression models, Gschlöβl and Czado [21] extend this to spatial Poisson regression models and Frühwirth-Schnatter and Frühwirth [15] show that the method is feasible for models involving other discrete-valued observations such as binary and multinomial data. The method involves introducing two sequences of latent variables through data augmentation. The first sequence serves to eliminate the non-linearity from the regression analysis but the non-normality of the error term remains. The error term can then be approximated by a mixture of Normal distributions to remove the non-normality. The component indicator of this mixture forms the second sequence of latent variables. However, a disadvantage of this method is that the number of latent variables introduced via the first sequence can be very high. For example, $y_i + 1$ latent variables are needed for each observation y_i in the Poisson model case. This means that the method is only really useful for data with small counts. Frühwirth-

Schnatter *et al.* [16] propose an improved version of auxiliary mixture sampling for count data, binomial data and multinomial data which involves a reduced number of latent variables. For example, at most two latent variables are introduced for each observation instead of $y_i + 1$ for the Poisson model.

Much work in the area of latent Gaussian models has been carried out by Rue, for example see [43], [44] and [45]. One of the most well known methods for tackling the problem outlined in this chapter is found in Rue *et al.* [45]. They propose using a Gaussian approximation to the Poisson regression model. However, when the observed counts are small the Poisson term is no longer approximated well by a Gaussian term so the method runs into problems.

In section 4.5 we develop a further improvement on the auxiliary mixture sampling method for the Poisson regression model. It further reduces the number of latent variables in the sense that it only requires one sequence of them, namely the component indicator variables for the mixture. We begin the section by explaining the method of Frühwirth-Schnatter *et al.* [16] in more detail highlighting how it differs from our method. We then proceed to present the method first for the univariate example of section 4.1.1 and then show that it can be extended to the multivariate case.

4.3 Generalisation of the auxiliary variable method

Suppose that we write our non-standard density as

$$p(x \mid y) \propto f(x)h(x)$$

where $f(x)$ is thought of as 'nice' and $h(x)$ is thought of as the 'nuisance' part forcing $p(x \mid y)$ have a non-standard form. Suppose now that we introduce an auxiliary variable u such that

$$p(x \mid u, y) \propto f(x)h(x)q(u \mid x) \tag{4.3}$$

Now our problem becomes finding $q(u \mid x)$ which in some sense 'removes' the $h(x)$ making $p(x \mid u, y)$ have a recognised distributional form, that of a Normal distribution in our case.

A possible way to approach this is to choose a density $\psi(u)$ which is easy to generate from and then either scale it by some function $g(x)$ or shift its location by some $g(x)$ to give us $q(u \mid x)$. We can find specific examples of both of these approaches in [11] which are described in sections 4.3.1 and 4.3.2 respectively.

4.3.1 Scaling

Suppose

$$q(u \mid x) = \left|\frac{1}{g(x)}\right| \psi\left(\frac{u}{g(x)}\right) \tag{4.4}$$

for some density $\psi(\cdot)$. Putting this into equation (4.3) gives

$$p(x \mid u, y) \propto f(x)\frac{h(x)}{g(x)}\psi\left(\frac{u}{g(x)}\right) \tag{4.5}$$

and our problem is now to find a $\psi(\cdot)$ and a $g(x)$ which makes $p(x \mid u, y)$ take a recognised distributional form. More specifically we want to find a $g(x)$ which cancels out the 'nuisance' part of $h(x)$.

Example

Suppose we take $\psi(\cdot)$ to be a Uniform(0,1) density and let $g(x) = h(x)$ which leads to

$$q(u \mid x) = \frac{1}{h(x)} \begin{cases} 1 & \text{if } 0 < u/h(x) < 1 \\ 0 & \text{otherwise} \end{cases}$$

and

$$p(x \mid u, y) \propto f(x) \begin{cases} 1 & \text{if } u \in (0, h(x)) \\ 0 & \text{otherwise} \end{cases}$$

i.e. $p(x \mid u, y)$ is a truncated version of $f(x)$ restricted to the set $\{x : h(x) \geq u\}$.

4.3.2 Location shift

An alternative way to specify $q(u \mid x)$ is

$$q(u \mid x) = \psi(u - g(x))$$

for some density $\psi(\cdot)$ and putting this into equation (4.3) gives

$$p(x \mid u, y) \propto f(x)h(x)\psi(u - g(x))$$

Example

Suppose we want to generate from

$$p(x) \propto f(x)h(x)$$

where $f(x)$ is a density of known form and $h(x) = e^{-e^x}$. Suppose now we use an Exponential density with mean 1 for $\psi(\cdot)$ and set $g(x) = e^x$. It follows that

$$q(u \mid x) = \begin{cases} e^{-u+e^x} & \text{if } u \geq e^x \\ 0 & \text{otherwise} \end{cases}$$

and therefore

$$p(x \mid u, y) \propto \begin{cases} f(x) & \text{if } x \leq log(u) \\ 0 & \text{otherwise} \end{cases}$$

i.e. $p(x \mid u, y)$ is a truncated version of $f(x)$ restricted to the set $\{x : x \leq log(u)\}$.

4.4 Adapting the auxiliary variable method

We now turn our attention to the scale construction of $q(x \mid u)$ given in (4.4) and consider whether there are alternative combinations of $\psi(\cdot)$ and a $g(x)$ which would not result in truncation. We can also note here that the method of this section can be applied to other latent Gaussian models, not just the Poisson case.

4.4.1 Requirements for $\psi(\cdot)$ and $g(x)$

We can rewrite our problem defined in (4.2) as wanting to make

$$p(x \mid y) \propto exp\Big\{ -\frac{1}{2\sigma^2}(x - \mu_*)^2 \Big\} e^{kx} e^{-e^x} \tag{4.6}$$

take the form of a Normal distribution for some $k \in \mathbb{R}$ where $\mu_* = \mu + \sigma^2(y - k)$. We introduce k here simply because we can choose its value freely which may help in finding an efficient sampling scheme. Using equation (4.5) it follows that

$$p(x \mid u, y) \propto exp\Big\{ -\frac{1}{2\sigma^2}(x - \mu_*)^2 \Big\} \frac{e^{kx} e^{-e^x}}{g(x)} \psi\Big(\frac{u}{g(x)}\Big). \tag{4.7}$$

One option would be to choose $g(x)$ and $\psi(\cdot)$ in such a way that $p(x \mid u, y)$ takes the form of a Normal density. We now restrict our attention to this case which means that we require

$$\psi\Big(\frac{u}{g(x)}\Big) \propto e^{Q(x)} \tag{4.8}$$

and

$$\frac{e^{kx}e^{-e^x}}{g(x)} \propto e^{Q(x)} \tag{4.9}$$

where $Q(x)$ is any quadratic in x for which the coefficient of x^2 is negative. Both of these parts could then be absorbed into the Normal density. We can also note that since $p(x \mid y)$ involves a Normal density, there will be some values of x far away from the peak for which it will have virtually no probability. It would not be necessary for equations (4.8) and (4.9) to hold for such values of x.

In section 4.4.2 we define what this necessary range of x is and in section 4.4.3 we consider possible combinations of density $\psi(\cdot)$ and function $g(x)$ which would ensure equations (4.8) and (4.9) hold for this range.

4.4.2 The plausible range of x values

In this section we want to find the values of x far away from the peak of $p(x \mid y)$ which have virtually no probability. We begin by looking at a sketch of $p(x \mid y)$ for some chosen values of μ, σ, y and k which is shown in Figure 4.1. The x-axis shows $\widehat{x}$ which is the value of x that maximises $p(x \mid y)$ as well as $\widehat{x} - r_0$ and $\widehat{x} + r_1$ which mark the boundaries beyond which x has virtually no probability. We let this 'small probability' area take the maximum value of $\epsilon \approx 0$ and more specifically $\epsilon/2$ in each of the tails. Our problem now is to find the values of $\widehat{x}$, $r_0 > 0$ and $r_1 > 0$.

Finding $\widehat{x}$

To find $\widehat{x}$ we use the Newton-Raphson method which is an algorithm for finding an approximate solution to the equation $\gamma(x) = 0$. Suppose we have some current approximation x_n, then we can find a better one x_{n+1} using

$$x_{n+1} = x_n - \frac{\gamma(x_n)}{\gamma'(x_n)}. \tag{4.10}$$

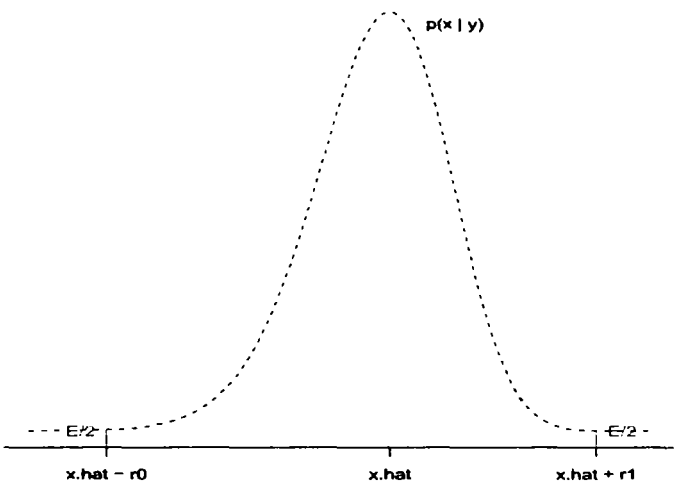


Figure 4.1: Illustrating the plausible range of x

This will then converge to the solution provided the initial guess x_0 was good. To find $\widehat{x}$ we want to find the maximum of $p(x \mid y)$, which will also be the maximum of $\log p(x \mid y)$. We therefore define

$$\begin{aligned}\gamma(x) &= \frac{d}{dx} log p(x \mid y) \\ &= \frac{d}{dx}\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 + kx - e^x \Big\} \\ &= -\frac{x}{\sigma^2} + \frac{\mu_*}{\sigma^2} + k - e^x\end{aligned}$$

and this gives $\gamma'(x) = -\frac{1}{\sigma^2} - e^x$. We then choose initial value $x_0 = \mu_*$ and proceed to find $\widehat{x}$ using the iterative formula (4.10).

Finding r_0 and r_1: overview

We begin by writing the information in Figure 4.1 more formally as

$$\int_{-\infty}^{\widehat{x}-r_0} p(x \mid y)\, dx \quad \leq \quad \epsilon_0 = \frac{\epsilon}{2} \tag{4.11}$$

$$\int_{\widehat{x}+r_1}^{\infty} p(x \mid y)\, dx \quad \leq \quad \epsilon_1 = \frac{\epsilon}{2} \tag{4.12}$$

where

$$p(x \mid y) = \frac{\eta(x)}{\int \eta(x) dx} \tag{4.13}$$

and

$$\eta(x) = exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} e^{kx} e^{-e^x}.$$

Our aim is then to find the upper bounds ϵ_0 and ϵ_1 in terms of r_0 and r_1 respectively and set each one equal to $\epsilon/2$ to determine the r-values. However, the presence of $\int \eta(x)dx$ in (4.13) makes it difficult to find the upper bounds directly. To proceed, we eliminate $\int \eta(x)dx$ by dividing by $p(\widehat{x} \mid y)$ as follows

$$\begin{aligned} \int_{-\infty}^{\widehat{x}-r_0} \frac{p(x \mid y)}{p(\widehat{x} \mid y)}\, dx &\leq \epsilon_0^* \\ \int_{\widehat{x}+r_1}^{\infty} \frac{p(x \mid y)}{p(\widehat{x} \mid y)}\, dx &\leq \epsilon_1^* \end{aligned}$$

and find the upper bounds ϵ_0^* and ϵ_1^* which are easier to compute. We can then rewrite (4.11) and (4.12) as

$$\begin{aligned} \int_{-\infty}^{\widehat{x}-r_0} p(x \mid y)\, dx &\leq \epsilon_0^*\, \overline{p}(\widehat{x} \mid y) = \frac{\epsilon}{2} \\ \int_{\widehat{x}+r_1}^{\infty} p(x \mid y)\, dx &\leq \epsilon_1^*\, \overline{p}(\widehat{x} \mid y) = \frac{\epsilon}{2} \end{aligned} \tag{4.14}$$

where $\overline{p}(\widehat{x} \mid y)$ is the upper bound for $p(\widehat{x} \mid y)$ which is also easy to compute. This procedure is very similar whether we are finding r_0 or r_1 so we now explain it in more detail for only one of them, namely r_1, but will explain where it differs for r_0.

Finding r_0 and r_1: details

We begin by writing

$$\frac{p(x \mid y)}{p(\widehat{x} \mid y)} = \frac{\eta(x)}{\eta(\widehat{x})} = \frac{exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 + kx - e^x \Big\}}{exp\Big\{ -\frac{1}{2\sigma^2}(\widehat{x}-\mu_*)^2 + k\widehat{x} - e^{\widehat{x}} \Big\}}$$

and since $e^x > 0$ for all x we know that an upper bound can be given by

$$\frac{p(x \mid y)}{p(\widehat{x} \mid y)} \leq exp\Big\{ \frac{1}{2\sigma^2}(\widehat{x}-\mu_*)^2 - k\widehat{x} + e^{\widehat{x}} \Big\} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 + kx \Big\}.$$

Note now that

$$-\frac{1}{2\sigma^2}(x-\mu_*)^2 + kx = -\frac{1}{2\sigma^2}(x-\mu_{**})^2 + k\mu + \sigma^2 ky - \frac{1}{2}\sigma^2 k^2$$

where $\mu_{**} = \mu + \sigma^2 y$ which gives us

$$\frac{p(x \mid y)}{p(\widehat{x} \mid y)} \leq exp\Big\{ \frac{1}{2\sigma^2}(\widehat{x}-\mu_*)^2 - k\widehat{x} + e^{\widehat{x}} \Big\} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_{**})^2 + k\mu + \sigma^2 ky - \frac{1}{2}\sigma^2 k^2 \Big\}$$

and so we can write

$$\int_{\widehat{x}+r_1}^{\infty} \frac{p(x \mid y)}{p(\widehat{x} \mid y)} dx \leq exp\Big\{\frac{1}{2\sigma^2}(\widehat{x}-\mu_*)^2 - k\widehat{x} + e^{\widehat{x}} + k\mu + \sigma^2 ky - \frac{1}{2}\sigma^2 k^2\Big\} \times \int_{\widehat{x}+r_1}^{\infty} exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_{**})^2\Big\} dx. \quad (4.15)$$

Note that if we were finding r_0 then the limits on the above integrals would be $-\infty$ and $\widehat{x} - r_0$. Suppose now that $X \sim N(\mu_{**}, \sigma^2)$ where the corresponding probability density function and cumulative distribution functions are as follows

$$\phi(x) = \frac{1}{\sigma\sqrt{2\pi}} exp\Big\{\frac{1}{\sigma^2}(x-\mu_{**})^2\Big\}$$
$$\Phi_{\mu_{**},\sigma}(x) = \int_{-\infty}^{x} \phi(u) du$$

and note that

$$1 - \Phi_{\mu_{**},\sigma}(x) = \int_{x}^{\infty} \phi(u) du$$
$$\Phi_{\mu_{**},\sigma}(x) = \Phi\Big(\frac{x-\mu_{**}}{\sigma}\Big).$$

where $\Phi(\cdot)$ is that c.d.f. for the standard Normal distribution. We now use the above to rewrite (4.15) as

$$\int_{\widehat{x}+r_1}^{\infty} \frac{p(x \mid y)}{p(\widehat{x} \mid y)} dx \leq exp\Big\{\frac{1}{2\sigma^2}(\widehat{x}-\mu_*)^2 - k\widehat{x} + e^{\widehat{x}} + k\mu + \sigma^2 ky - \frac{1}{2}\sigma^2 k^2\Big\} \times \sigma\sqrt{2\pi}\Big[1 - \Phi\Big(\frac{\widehat{x}+r_1-\mu_{**}}{\sigma}\Big)\Big] = \epsilon_1^*. \quad (4.16)$$

Note that for calculating ϵ_0^* the contents of the square bracket would be changed to $\Phi\big(\frac{\widehat{x}-r_0-\mu_{**}}{\sigma}\big)$. The next step is to find $\overline{p}(\widehat{x} \mid y)$, the upper bound for $p(\widehat{x} \mid y)$. This can be written as

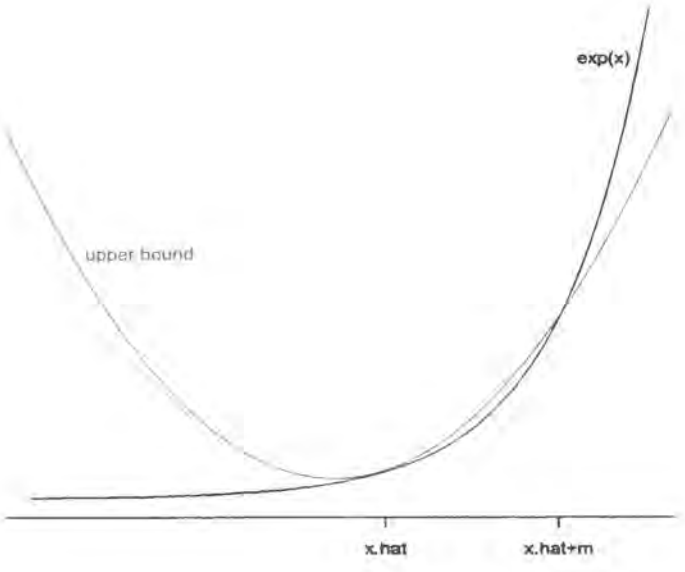
$$\overline{p}(\widehat{x} \mid y) = \frac{\eta(\widehat{x})}{\int \underline{\eta}(x) dx}$$

where $\underline{\eta}(x)$ is a lower bound for $\eta(x)$. Since we know that

$$\eta(x) = exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_*)^2\Big\} e^{kx} e^{-e^x},$$

it is clear that a lower bound for $\eta(x)$ can be obtained by finding an upper bound for e^x. One such upper bound for e^x is highlighted in Figure 4.2. It is obtained by

Figure 4.2: Illustrating an upper bound for e^x



fitting a quadratic (shown in red) above the e^x curve touching it at $\widehat{x}$. We define this quadratic as $\xi(x) = ax^2 + bx + c$ for some $a, b, c \in \mathbb{R}$. This means that at $\widehat{x}$, the curves are equal

$$\xi(\widehat{x}) = a\widehat{x}^2 + b\widehat{x} + c = e^{\widehat{x}}, \tag{4.17}$$

the first derivatives are equal

$$\xi'(\widehat{x}) = 2a\widehat{x} + b = e^{\widehat{x}}, \tag{4.18}$$

and the second derivative of $\xi(x)$ is greater than the second derivative of e^x

$$\xi''(\widehat{x}) = 2a > e^{\widehat{x}}. \tag{4.19}$$

We can then use (4.17) to (4.19) to find b and c in terms of a and $\widehat{x}$ so that a is the only unknown. More formally,

$$\xi(x) = ax^2 + (e^{\widehat{x}} - 2\widehat{x}a)x + e^{\widehat{x}}(1 - \widehat{x}) + \widehat{x}^2 a$$

where $a > \frac{1}{2}e^{\widehat{x}}$. We can also see from Figure 4.2 that there will come a point, namely $\widehat{x} + m$, at which $\xi(x)$ crosses e^x meaning that $\xi(x)$ is no longer the upper bound.

This is true for whichever value of a we choose. Also, since $\xi(\widehat{x}+m) = e^{\widehat{x}+m}$ we know that

$$a = \frac{(-m-1+e^m)e^{\widehat{x}}}{m^2} > \frac{1}{2}e^{\widehat{x}}$$

and our value of a can be obtained from whatever value of $m > 0$ we choose. We can now define our lower bound of $\eta(x)$ as

$$\underline{\eta}(x) = \begin{cases} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 + kx - \xi(x)\Big\} & \text{for } x \le \widehat{x}+m \\ 0 & \text{for } x > \widehat{x}+m. \end{cases}$$

Since we can write

$$-\frac{1}{2\sigma^2}(x-\mu_*)^2 + kx - \xi(x) = -\frac{1}{2\widetilde{\sigma}^2}(x-\widetilde{\mu})^2 + C$$

where

$$\widetilde{\mu} = \tfrac{\mu+\sigma^2(y-e^{\widehat{x}}+2\widehat{x}a)}{1+2a\sigma^2}$$
$$\widetilde{\sigma}^2 = \tfrac{\sigma^2}{1+2a\sigma^2},$$
$$C = \tfrac{1}{2}\left(\left(\tfrac{\widetilde{\mu}}{\widetilde{\sigma}}\right)^2 - \left(\tfrac{\mu}{\sigma}\right)^2 - \sigma^2(y-k)^2\right) - (y-k)\mu - e^{\widehat{x}}(1-\widehat{x}) - \widehat{x}^2 a$$

then the lower bound can be written as

$$\underline{\eta}(x) = \begin{cases} exp\Big\{ -\frac{1}{2\widetilde{\sigma}^2}(x-\widetilde{\mu})^2 + C\Big\} & \text{for } x \le \widehat{x}+m \\ 0 & \text{for } x > \widehat{x}+m. \end{cases}$$

We can therefore write the integral of this lower bound as

$$\begin{aligned} \int \underline{\eta}(x)dx &= exp\{C\}\int_{-\infty}^{\widehat{x}+m} exp\Big\{ -\frac{1}{2\widetilde{\sigma}^2}(x-\widetilde{\mu})^2\Big\}dx \\ &= exp\{C\}\,\widetilde{\sigma}\sqrt{2\pi}\,\Phi\Big(\frac{\widehat{x}+m-\widetilde{\mu}}{\widetilde{\sigma}}\Big) \end{aligned}$$

where $\Phi(\cdot)$ is that c.d.f. for the standard Normal distribution. It therefore follows that

$$\begin{aligned} \overline{p}(\widehat{x} \mid y) &= \frac{\eta(\widehat{x})}{\int \underline{\eta}(x)dx} \\ &= \frac{exp\Big\{ -\frac{1}{2\sigma^2}(\widehat{x}-\mu_*)^2 + k\widehat{x} - e^{\widehat{x}}\Big\}}{exp\{C\}\,\widetilde{\sigma}\sqrt{2\pi}\,\Phi\Big(\frac{\widehat{x}+m-\widetilde{\mu}}{\widetilde{\sigma}}\Big)} \end{aligned} \tag{4.20}$$

Note that there are still some unknown constants, namely m and k, which we are free to choose in order to minimise r_1 (*and* r_0) thereby reducing the range for which equations (4.8) and (4.9) must hold.

From equations (4.16) and (4.20), we now know ϵ_1^* (in terms of r_1) and $\overline{p}(\widehat{x} \mid y)$ so, for any given value of ϵ, we can find r_1 using the right-hand side of the inequality in equation (4.14). More specifically,

$$\begin{aligned}\frac{\epsilon}{2} &= \epsilon_1^* \overline{p}(\widehat{x} \mid y) \\ &= \frac{exp\Big\{k\mu + \sigma^2 ky - \frac{1}{2}\sigma^2 k^2 - C\Big\}\sigma\left[1 - \Phi\Big(\frac{\widehat{x}+r_1-\mu_{**}}{\sigma}\Big)\right]}{\widetilde{\sigma}\,\Phi\Big(\frac{\widehat{x}+m-\widetilde{\mu}}{\widetilde{\sigma}}\Big)}\end{aligned}$$

which leads to

$$r_1 = \sigma\Phi^{-1}(1-p) + \mu_{**} - \widehat{x}$$

where

$$p = \frac{\epsilon\widetilde{\sigma}\Phi\Big(\frac{\widehat{x}+m-\widetilde{\mu}}{\widetilde{\sigma}}\Big)}{2exp\Big\{k\mu + \sigma^2 ky - \frac{1}{2}\sigma^2 k^2 - C\Big\}\sigma} \in [0,1].$$

Note that we find r_0 in a similar way using ϵ_0^* instead of ϵ_1^* before we simplify. This gives us

$$r_0 = -\sigma\Phi^{-1}(p) - \mu_{**} + \widehat{x}.$$

We can also note here that $\Phi^{-1}(1-p) = -\Phi^{-1}(p)$ meaning that we can calculate r_0 from r_1 as follows

$$r_0 = r_1 - 2(\mu_{**} - \widehat{x}).$$

Since we need r_0 to be greater than 0 then we must choose m and k in such a way that r_1 is greater than the maximum of 0 and $2(\mu_{**} - \widehat{x})$. Once we have calculated $\widehat{x}$, r_0 and r_1, it follows that the plausible range of x values is

$$\Omega = [\widehat{x} - r_0, \widehat{x} + r_1].$$

4.4.3 Choosing $\psi(\cdot)$ and $g(x)$

We first consider defining $\psi(\cdot)$ and $g(x)$ such that equation (4.8) holds. Recall that for $x \in \Omega$ we require $\psi\Big(\frac{u}{g(x)}\Big) \propto e^{Q(x)}$ for some quadratic $Q(x)$ with negative coefficient of x^2. We begin by noting some restrictions on the domain of $\psi(\cdot)$.

Domain of $\psi(\cdot)$

Suppose that we take ψ to be a Beta(a, b) density such that

$$\psi(t) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} t^{a-1}(1-t)^{b-1}, \ t \in [0,1]$$

then to ensure equation (4.8) holds we must set $b = 1$ and $g(x) = e^{Q^+(x)}$. However, the definition of $\psi(\cdot)$ means that $\frac{u}{g(x)} \in [0,1]$ which would lead to a truncation in generating from $p(x \mid u, y)$ and, as we have already discussed, generating from a truncated multivariate Normal distribution is difficult. We can generalise this to say that when defining $\psi(t)$, the domain of t must be $\mathbb{R}$ or $\mathbb{R}^+$ to prevent it from resulting in a truncated distribution for $p(x \mid u, y)$.

Ensuring equation (4.8) holds

Suppose that

$$\begin{aligned} \psi(t) &\propto exp\{-t^{1/n}\}, \ t \in \mathbb{R}^+, n \in \mathbb{R}^+ \\ \psi\left(\frac{u}{g(x)}\right) &\propto exp\Big\{-\left(\frac{u}{g(x)}\right)^{1/n}\Big\} \end{aligned}$$

and we can note that if $n = 1$, then $\psi(\cdot)$ takes the form of an Exponential density with mean 1. To ensure that equation (4.8) holds we must set $g(x) = \left[\frac{1}{Q^+(x)}\right]^n$ where $Q^+(x)$ is any positive quadratic in x. It is therefore relatively straightforward to ensure that $\psi\left(\frac{u}{g(x)}\right) \propto e^{Q(x)}$ but now we need to further define $g(x)$ to ensure that equation (4.9) holds which is a more difficult task.

Ensuring equation (4.9) holds

Recall that we want to find $k \in \mathbb{R}$ and $g(x) = \left[Q^+(x)\right]^{-n}$ such that $\frac{e^{kx}e^{-e^x}}{g(x)} \propto e^{Q(x)}$ for all $x \in \Omega$. Suppose that we now let $Q^+(x) = \alpha x^2 + \beta x + \delta$ and rewrite equation (4.9) as

$$\phi(x) \stackrel{def}{=} kx - e^x + nlog(\alpha x^2 + \beta x + \delta) \propto Q(x).$$

We proceed by finding the value of x which maximises $\phi(x)$ and fit a quadratic $\zeta(x)$ through this point. We then quantify how close $\zeta(x)$ and $\phi(x)$ are using

$$I = \int_\Omega \mid \phi(x) - \zeta(x) \mid \ dx$$

and use R function `optim()` to find the values of α, β, δ and n which minimise I.

Numerical example

Suppose that $\mu = log(2)$, $\sigma = 1$ and $y = 3$ such that

$$\begin{aligned} f(x) &= exp\Big\{ -\frac{1}{2}\big(x - log(2)\big)^2 \Big\} \\ h(x) &= (e^x)^3 e^{-e^x}. \end{aligned}$$

and equation (4.7) becomes

$$p(x \mid u, y) \propto exp\Big\{ -\frac{1}{2}\big(x - [log(2) + (3-k)]\big)^2 \Big\} \frac{e^{kx} e^{-e^x}}{g(x)} \psi\Big(\frac{u}{g(x)}\Big)$$

We first find Ω, the plausible range of x values, using the method described in section 4.4.2. Using the R function `optim()` to find the values of m and k which minimise this range we obtain

$$\Omega = [-6.734657, 8.721104].$$

If we take $\psi(t) \propto exp\{-t^{1/n}\}$ then we must let $g(x) = [Q^+(x)]^{-n}$ to guarantee (4.8) holds. Using the above method for ensuring (4.9) holds, we find $g(x)$ and $\zeta(x)$ to be

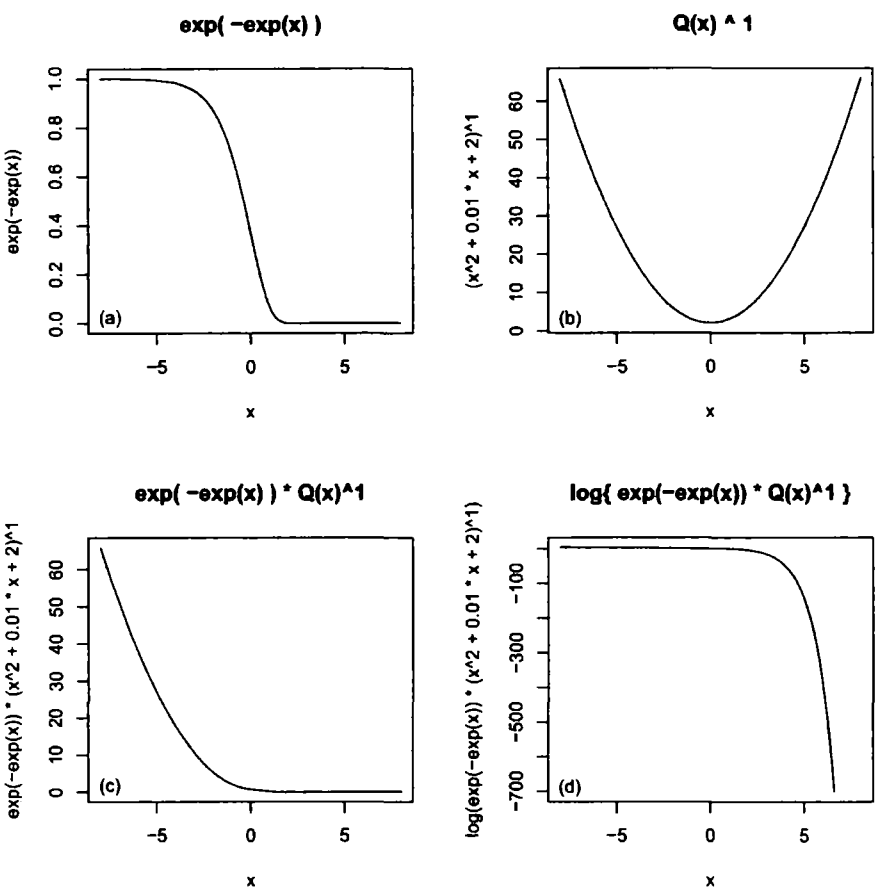
$$\begin{aligned} g(x) &= (2.183364x^2 + 0.8487513x + 1.440820)^{-2.656443} \\ \zeta(x) &= -0.1107425x^2 - 1.101230x + 2.293468 \end{aligned}$$

which lead to a minimum value of $I = 6266.303$. We therefore conclude that the quadratic $\zeta(x)$ is a poor approximation for $\phi(x)$ and it doesn't seem possible to find a $\phi(x) \propto Q(x)$ using the method described in this section.

4.4.4 The problem with the method

The difficulty arises because e^{-e^x} has an awkward asymmetrical shape shown in Figure 4.3(a) which makes it hard to find a $g(x)$ to 'cancel' it out. We know that we would like $g(x)^{-1}$ to take the form $[Q^+(x)]^n$ which has a symmetrical shape. Figure 4.3(b) shows $g(x)^{-1}$ for $n = 1$ and some $Q^+(x)$. Figure 4.3(c) shows $e^{-e^x} g(x)^{-1}$ and Figure 4.3(d) shows the same thing on log-scale. On first glance it could appear that (c) shows the tail of a Normal density curve but when we look on log scale it is clear that (d) is not part of a quadratic as it would need to be for this approach to work.

Figure 4.3: Illustrating the problem with the method



Does adding another auxiliary variable improve things?

It is possible to introduce another auxiliary variable w which changes the problem in such a way that $g(x)$ has a symmetrical shape to cancel out instead of the asymmetrical one, which we hope could improve things. Suppose

$$\begin{aligned} w &\sim Poisson(e^{-x}) \\ p(w \mid x) &\propto e^{-wx} e^{-e^{-x}} \end{aligned}$$

and this leads to

$$\begin{aligned} p(x \mid u,y,w) &\propto p(x \mid u,y)p(w \mid x) \\ &\propto exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 - wx \Big\} \frac{e^{kx}e^{-(e^x+e^{-x})}}{g(x)} \psi\Big(\frac{u}{g(x)}\Big) \\ &\propto exp\Big\{ -\frac{1}{2\sigma^2}(x-\widetilde{\mu}_*)^2 \Big\} \frac{e^{-(e^x+e^{-x})}}{g(x)} \psi\Big(\frac{u}{g(x)}\Big) \end{aligned}$$

for some constant $\widetilde{\mu}_*$ depending on k, w and y. Note that $cosh(x) = \frac{1}{2}(e^x + e^{-x})$ so our problem now becomes finding

$$\psi\Big(\frac{u}{g(x)}\Big) \propto e^{Q(x)}$$

and

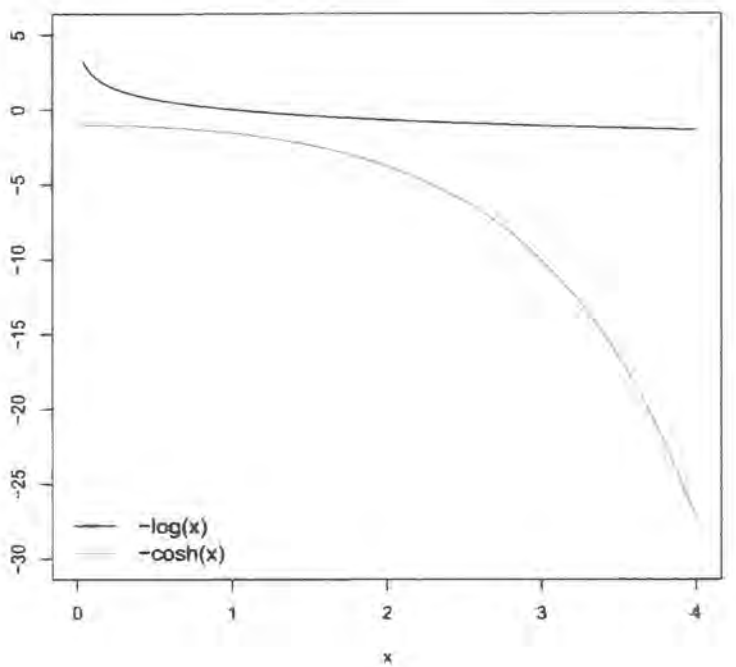
$$\frac{e^{-2cosh(x)}}{g(x)} \propto e^{Q(x)}$$

If we take $\psi(t) \propto exp\{-t^{1/n}\}$ as we have discussed previously, then we require $g(x) = [Q^+(x)]^{-n}$ which is symmetric. Although $g(x)$ now has something symmetric to cancel out, the main problem is that $e^{-2cosh(x)}$ goes to zero too quickly for $g(x)$ to be $[Q^+(x)]^{-n}$. To illustrate this we look at the behaviour of the logarithms of $e^{-2cosh(x)}$ and $[Q^+(x)]^{-n}$. Firstly we note that $-2cosh(x)$, by definition, tends to $-\infty$ exponentially fast as it moves away from 0. We now consider $-nlog[Q^+(x)]$ which, for large x, behaves like $-nlog(ax^2)$ for some $a \in \mathbb{R}$. We can therefore think of the behaviour of $-nlog[Q^+(x)]$ as essentially the same as the behaviour of $-nlog(a) - 2nlog(x)$. Now, although $-log(x)$ does tend towards $-\infty$ as it moves away from 0, it does so very slowly in comparison to $cosh(x)$. This is illustrated in Figure 4.4. This means that setting $g(x) = [Q^+(x)]^{-n}$ wouldn't be sufficient to compensate for the awkward behaviour of $e^{-2cosh(x)}$.

Our options now are to try to fix the problem by making a better choice for $\psi(\cdot)$ or by using an approach other than scale construction when introducing the auxiliary variable. However, intuition says that this might not be possible. In the overall set up of the problem, recall that we have

$$p(x \mid y) \propto exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} e^{kx}e^{-e^x}$$

Figure 4.4: Illustrating the behaviour of -log(x) and -cosh(x)



and we introduce the auxiliary variable u as follows

$$p(x \mid y, u) \propto exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} e^{kx} e^{-e^x} p(u \mid x) \tag{4.21}$$

such that we want $p(x \mid y, u)$ to take the form of a Normal density. We therefore need

$$p(u \mid x) = \frac{1}{\delta(x)} exp\big\{ a(u)x^2 + b(u)x + c(u) \big\}$$

where a, b and c are some functions of u and $\delta(x)$ is the normalising constant which needs to 'cancel out' the nuisance part of $p(x \mid y)$. This means that we would require $\delta(x)$ to behave similarly to e^{-e^x} (or $e^{-2cosh(x)}$ if we introduce the extra auxiliary variable to make things symmetric). The question now becomes whether or not we can choose a, b and c for this to be the case. Note that

$$\delta(x) \;=\; \int exp\big\{ a(u)x^2 + b(u)x + c(u) \big\} du$$

and we now look at the basic behaviour of $\delta(x)$ and first consider the case for which $x > 0$. Suppose we let A_0 be some interval of positive length on which $a(u)$ and

$b(u)$ are bounded below and let α_0 be the infimum of $\{a(\cdot), b(\cdot)\}$ on that interval. We then have that

$$\begin{aligned}\int exp\{a(u)x^2 + b(u)x + c(u)\}du &\geq \int_{A_0} exp\{a(u)x^2 + b(u)x + c(u)\}du \\ &\geq \int_{A_0} exp\{\alpha_0 x^2 + \alpha_0 x + c(u)\}du \\ &= exp\{\alpha_0(x^2 + x)\}\int_{A_0} exp\{c(u)\}du\end{aligned}$$

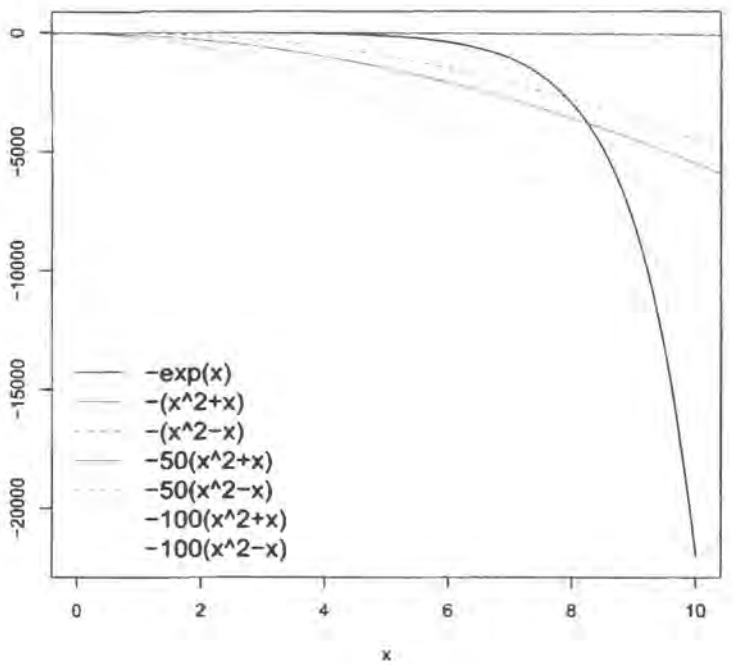
so we know that $\delta(x)$ can't go to 0 faster than $exp\{\alpha_0(x^2+x)\}$ as $x \to \infty$. Suppose now that $x < 0$ and we let A_1 be some interval of positive length on which $a(u)$ and $-b(u)$ are bounded below. Furthermore, we let α_1 be the infimum of $\{a(\cdot), -b(\cdot)\}$ on that interval. We then have

$$\begin{aligned}\int exp\{a(u)x^2 + b(u)x + c(u)\}du &\geq \int_{A_1} exp\{a(u)x^2 - b(u)x + c(u)\}du \\ &\geq \int_{A_1} exp\{\alpha_1 x^2 + \alpha_1 x + c(u)\}du \\ &= exp\{\alpha_1(x^2 + x)\}\int_{A_1} exp\{c(u)\}du\end{aligned}$$

so we know that $\delta(x)$ can't go to 0 faster than $exp\{\alpha_1(x^2 + x)\}$ as $x \to -\infty$. We then let $\alpha = min\{\alpha_0, \alpha_1\}$ and are able to conclude that $\delta(x)$ can't go to 0 faster than $exp\{\alpha(x^2 + |x|)\}$ as $|x| \to \infty$ which means that $\delta(x)$ can't behave similarly to $exp\{-exp\{x\}\}$. Firstly, if $\alpha > 0$ it follows that $exp\{\alpha(x^2 + |x|)\}$ tends to ∞ as $|x| \to \infty$ rather than the 0 we require. If $\alpha < 0$ then although $exp\{\alpha(x^2 + |x|)\}$ does tend to 0, it does so much slower than $exp\{-exp\{x\}\}$. In other words, $\alpha(x^2+|x|)$ goes to $-\infty$ much slower than $-exp\{x\}$ which is illustrated in Figure 4.5 for $\alpha = -1, -50$ and -100.

This suggests that it is not possible to find an a, b and c such that $\delta(x)$ behaves similarly to e^{-e^x} or $e^{-2cosh(x)}$. It therefore seems that introducing $p(u \mid x)$ as in (4.21) is not possible, irrespective of our choice of $\psi(\cdot)$ or whether or not we use scale construction in defining $p(x \mid u)$. We now turn our attention to a different method of solving the problem, namely that of auxiliary mixture sampling which was introduced in section 4.2.

Figure 4.5: Illustrating the behaviour of $-exp\{x\}$ and $\alpha(x^2 + x)$



4.5 Auxiliary mixture approximation

We begin by describing the method of Frühwirth-Schnatter *et al.* [16] which is a recent approach to this type of problem. We do so in the context of the univariate example outlined in section 4.1.1. Recall that

$$\begin{aligned} Y &\sim Poisson(\lambda) \\ \lambda &= e^x \\ X &\sim N(\mu, \sigma^2) \end{aligned}$$

and we want to make the posterior distribution $p(x \mid y)$ take the form of a Normal distribution by introducing augmenting variables. The approach begins by letting y be the number of jumps of an unobserved Poisson process with intensity λ, having occurred in the time interval $0 \leq t \leq 1$. They then define τ_2^* to be the arrival time of the last jump before $t = 1$ and τ_1^* to be the interarrival time between the last

jump before and the first jump after $t = 1$. It then follows that for $y > 0$,

$$\begin{aligned} \tau_1^* &= \frac{\xi_1}{\lambda}, \qquad \xi_1 \sim Exp(1) \\ \tau_2^* &= \frac{\xi_2}{\lambda}, \qquad \xi_2 \sim Gamma(y, 1) \end{aligned}$$

and note that for $y = 0$ we are only dealing with the equation involving τ_1^*. It is here that the first of the latent variables are introduced:

$$\tau = \begin{cases} (\tau_1^*, \tau_2^*) & \text{if } y > 0 \\ \tau_1^* & \text{if } y = 0. \end{cases}$$

We have now eliminated the non-linearity of the observation equation but the error term is still non-Normal. More specifically,

$$\begin{aligned} -log\tau_1^* &= log\lambda + \epsilon_1, \qquad \text{where } \epsilon_1 = -log\xi_1 \\ -log\tau_2^* &= log\lambda + \epsilon_2, \qquad \text{where } \epsilon_2 = -log\xi_2. \end{aligned}$$

so ϵ_1 is the negative logarithm of an Exponential random variable with rate 1 and ϵ_1 is the negative logarithm of a Gamma random variable with shape y and unit scale. This is where the second set of latent variables comes in. Frühwirth-Schnatter *et al.* [16] describe how the densities of both ϵ_1 and ϵ_2 can be approximated by Normal mixtures

$$\begin{aligned} p(\epsilon_1) &\approx \sum_{r_1=1}^{R_1} w_{r_1} f(\epsilon_1; m_{r_1}, s_{r_1}^2) \\ p(\epsilon_2) &\approx \sum_{r_2=1}^{R_2} w_{r_2} f(\epsilon_2; \widetilde{m}_{r_2}, \widetilde{s}_{r_2}^2) \end{aligned}$$

where

$$\sum_{r_1=1}^{R_1} w_{r_1} = \sum_{r_2=1}^{R_2} w_{r_2} = 1$$

and $f(\cdot; m_r, s_r^2)$ is a Normal density with mean m_r and standard deviation s_r for the r-th component. The second set of latent variables are the component indicators for these mixtures such that

$$r = \begin{cases} (r_1, r_2) & \text{if } y > 0 \\ r_1 & \text{if } y = 0. \end{cases}$$

The conditional posterior distribution $p(x \mid y, \tau, r)$ then takes a Normal distribution as follows

$$p(x \mid y, \tau, r) \propto p(x) f\big(- log\tau_1^*; log\lambda + m_{r_1}, s_{r_1}^2 \big) f\big(- log\tau_2^*; log\lambda + \widetilde{m}_{r_2}, \widetilde{s}_{r_2}^2 \big).$$

and the conditional distributions $p(\tau \mid y, x, r)$ and $p(r \mid y, x, \tau)$ are also easy to generate from. The method we are about to introduce differs from this one in the sense that the first set of auxiliary variables τ are not necessary but is similar in the sense that it uses one of the Normal mixture approximations. The elimination of the first set is of particular benefit when we consider the multivariate version of the problem. The current method requires a τ to be introduced for each observation meaning that for a large data set, a large number of latent variables are needed. Our method begins by rearranging the posterior distribution $p(x \mid y)$ to comprise a Normal density multiplied by something which can be approximated by a mixture of Normals. We now introduce the method in more detail, initially for the univariate example.

4.5.1 Univariate case

In this section we show that it is possible to find a very good approximation to $p(x \mid y)$ which takes the form of a Normal density. We begin by using (4.6) with $k = 1$ to express

$$p(x \mid y) \propto exp\Big\{ -\frac{1}{2\sigma^2}(x - \mu_*)^2 \Big\} exp\Big\{ x - e^x \Big\} \tag{4.22}$$

where $\mu_* = \mu + \sigma^2(y - 1)$. We proceed by noting that $exp\{x - e^x\}$ can be approximated sufficiently closely by a mixture of Normal distributions.

Normal mixture approximation

Suppose that

$$\varphi(x) = exp\{x - e^x\} \tag{4.23}$$

which can be approximated, as shown by Frühwirth-Schnatter and Wagner [17], by a normal mixture of R components

$$\widehat{\varphi}(x) = \sum_{r=1}^{R} w_r f(x; m_r, s_r^2)$$

where $f(\cdot; m_r, s_r^2)$ is a Normal density with mean m_r and standard deviation s_r for the r-th component. Frühwirth-Schnatter and Frühwirth [15] use distance-based measures to fit the mixture to $exp\{x - e^x\}$ and evaluate the approximation. They find that the best performing approximation is for 10-components and is based on minimising the relative entropy between $\varphi(x)$ and $\widehat{\varphi}(x)$. The parameters (w_r, m_r, s_r^2) for $R = 10$ are given in Table 4.1.

Table 4.1: Parameter values for the 10-component Normal mixture approximation

r	1	2	3	4	5	6	7	8	9	10
w_r	0.004	0.040	0.168	0.147	0.125	0.101	0.104	0.116	0.107	0.088
m_r	-5.09	-3.29	-1.82	-1.24	-0.764	-0.391	-0.043	0.306	0.673	1.06
s_r^2	4.5	2.02	1.1	0.422	0.198	0.107	0.078	0.077	0.095	0.146

Introducing the auxiliary variable

We can now introduce the auxiliary variable r to (4.22) as follows

$$\begin{aligned} p(x, r \mid y) &\overset{\propto}{\sim} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} w_r f(x; m_r, s_r^2) \\ &\propto exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} \frac{w_r}{s_r} exp\Big\{ -\frac{1}{2s_r^2}(x-m_r)^2 \Big\} \end{aligned}$$

and summing over r then gives us

$$p(x \mid y) \overset{\propto}{\sim} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} \sum_{r=1}^{10} \frac{w_r}{s_r} exp\Big\{ -\frac{1}{2s_r^2}(x-m_r)^2 \Big\}.$$

Given one particular value of r, this leads to

$$\begin{aligned} p(x \mid y, r) &\overset{\propto}{\sim} exp\Big\{ -\frac{1}{2\sigma^2}(x-\mu_*)^2 \Big\} exp\Big\{ -\frac{1}{2s_r^2}(x-m_r)^2 \Big\} \\ &\propto exp\Big\{ -\frac{1}{2\widehat{\sigma}^2}(x-\widehat{\mu})^2 \Big\} \end{aligned} \tag{4.24}$$

where

$$\begin{aligned} \widehat{\mu} &= \frac{\mu_* s_r^2 + m_r \sigma^2}{s_r^2 + \sigma^2} \\ \widehat{\sigma}^2 &= \frac{s_r^2 \sigma^2}{s_r^2 + \sigma^2}. \end{aligned}$$

We note here that this can be thought of as a sequential Normal update where the mixture summaries m_r and ${s_r}^2$ can be considered as 'data' used to update the prior (μ_*, σ^2) to the posterior.

Generating the sample

To use a Gibbs sampling approach, we first generate a value of r (given an initial value for x) from discrete probability distribution $w_1^*, \ldots, w_{10}^*$ where

$$w_j^* = \frac{w_j f(x; m_j, s_j^2)}{\sum_{j=1}^{10} w_j f(x; m_j, s_j^2)}$$

is the probability of obtaining a value $r = j$. We then use this value of r to obtain $\widehat{\mu}$ and $\widehat{\sigma}^2$ and generate x from a Normal distribution with these parameters. We then use this most current value of x to generate another value of r which we use to generate another value of x, and so on.

How good is the approximation?

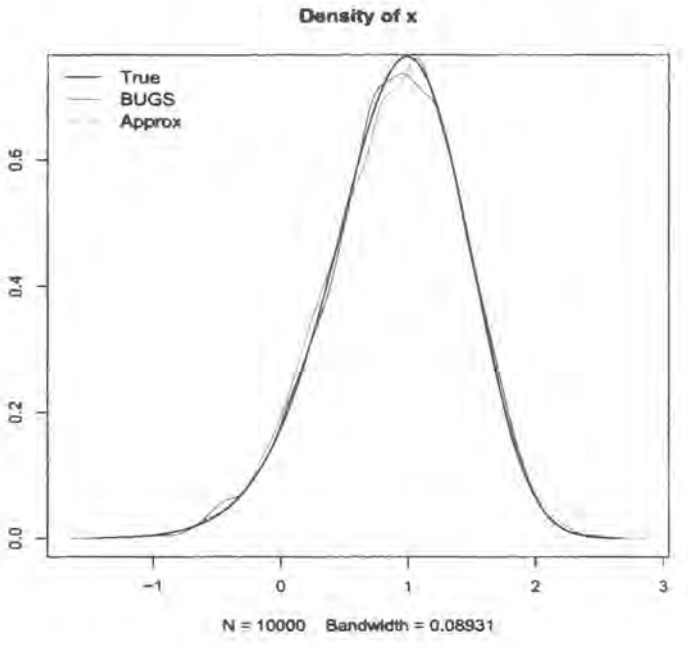
In order to see how well this auxiliary mixture approximation works for the univariate example given in section 4.1.1, we generate one sample using this method and one sample using `LinBUGS` (as described in section 3.3) then compare the two. We use the initial value $x = 1$, data $y = 3$ and parameter values $\mu = log(2)$ and $\sigma = 1$ and obtain samples of size 10,000. Figure 4.6 shows the posterior density obtained using a sample from `LinBUGS` in blue and from sampling using the improved auxiliary mixture approximation in red. We can see that they appear to be quite close to each other. Their summary values (shown in table 4.2) also confirm this since their means, standard deviations and quantiles are very close.

Table 4.2: Summary values for the univariate example

	mean	sd	2.5%	25 %	50 %	75 %	97.5 %
BUGS	0.899106	0.534976	-0.2433	0.5646	0.9274	1.2770	1.8460
Approx	0.889919	0.536570	-0.2244	0.5373	0.9286	1.2683	1.8502

We can also quantify the 'difference' between the two samples using relative entropy, as described in section 5.1.2. For our example, this is 0.00251 which can be thought of as a very small difference. Figure 4.6 also shows the true posterior density for the example which is close to the densities obtained using MCMC. We can therefore conclude that, for this example at least, our auxiliary mixture approximation not only works as well as `LinBUGS` but also produces an accurate estimate of

Figure 4.6: Univariate example showing accuracy of the approximation



the true posterior density. We can also note that it still works as well when different values of initial x, μ and σ are used. However, this is not the case as we increase the y value.

Auxiliary mixture approximation for large y

If we keep the same initial value $x = 1$ and parameter values $\mu = log(2)$ and $\sigma = 1$ but change the data to be $y = 10$ then a problem occurs with our auxiliary mixture approximation method, but not with the `LinBUGS` method. Figure 4.7 illustrates this.

The reason for the problem is due to the normal mixture not being a very good approximation for $exp\{x - e^x\}$ when x is large. The first plot in Figure 4.8 shows the true $exp\{x - e^x\}$ in black and the approximation to it in red for x between -10 and 10. On first glance the approximation appears to be very good. However, when we look at the same thing on log-scale (shown in the second plot) it is clear that the approximation begins to break down when $x \approx 10$.

A large value of y would increase the mean of $p(x \mid y, r)$ given by (4.24) thereby

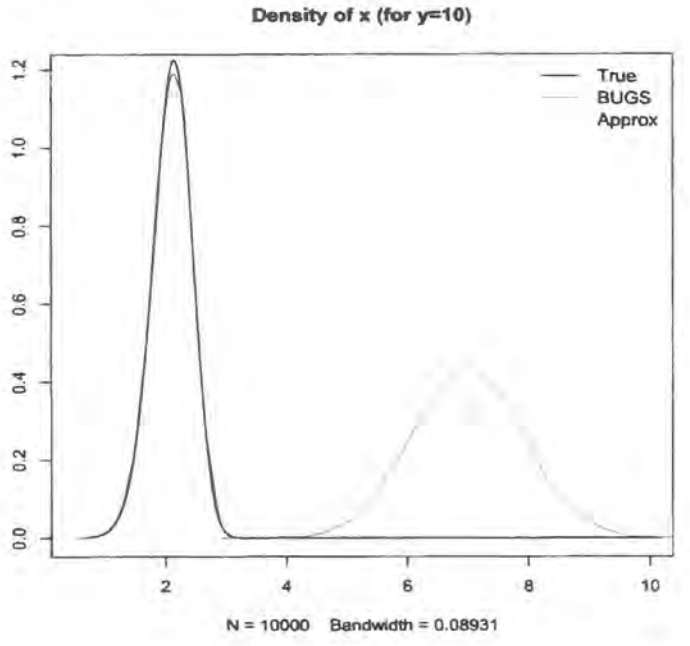


Figure 4.7: Breakdown of the auxiliary mixture approximation for large y

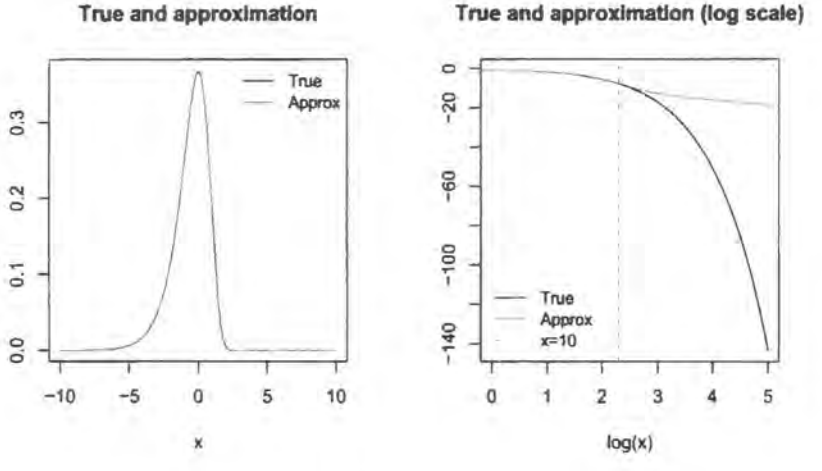


Figure 4.8: $exp\{x - e^x\}$ and its Normal mixture approximation

pushing the values of x generated more towards this area where the approximation breaks down. One solution to this problem is to find a way of shifting the area where the approximation is applied back into the region where it works. This is the

problem we now consider.

Improved auxiliary mixture approximation

In this section we present a way of shifting the area where the approximation is applied by introducing a second normal mixture distribution. First we note that our original problem defined in (4.2) can be expressed as

$$\begin{aligned} p(x \mid y) &\propto exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_{**})^2\Big\}exp\Big\{x-log(2)-e^{x-log(2)}\Big\} \\ &\qquad \times exp\Big\{x-log(2)-e^{x-log(2)}\Big\} \\ &\propto exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_{**})^2\Big\}\varphi_1\Big(x-log(2)\Big)\varphi_2\Big(x-log(2)\Big) \end{aligned}$$

where $\mu_{**} = \mu + \sigma^2(y-2)$ and $\varphi_i(\cdot)$ for $i = 1, 2$ is as defined in (4.23). Now we have

$$p(x \mid y) \overset{\propto}{\sim} exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_{**})^2\Big\}\sum_{r_1=1}^{10} w_{r_1} f(x; m^*_{r_1}, s^2_{r_1}) \sum_{r_2=1}^{10} w_{r_2} f(x; m^*_{r_2}, s^2_{r_2})$$

where $m^*_{r_i} = m_{r_i} + log(2)$ for $i = 1, 2$ and $f(\cdot)$ is as defined in section 4.5.1. It follows that

$$\begin{aligned} p(x \mid y, r_1, r_2) &\overset{\propto}{\sim} exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_{**})^2\Big\}exp\Big\{-\frac{1}{2s^2_{r_1}}\big(x-m^*_{r_1}\big)^2\Big\} \\ &\qquad \times exp\Big\{-\frac{1}{2s^2_{r_2}}\big(x-m^*_{r_2}\big)^2\Big\} \\ &\propto exp\Big\{-\frac{1}{2\sigma^2}(x-\mu_{**})^2\Big\}exp\Big\{-\frac{1}{2\widehat{s}^2}\big(x-\widehat{m}\big)^2\Big\} \\ &\propto exp\Big\{-\frac{1}{2\widetilde{\sigma}^2}\big(x-\widetilde{\mu}\big)^2\Big\} \end{aligned}$$

where

$$\begin{aligned} \widehat{m} &= \frac{m^*_{r_1}s^2_{r_2} + m^*_{r_2}s^2_{r_1}}{s^2_{r_1} + s^2_{r_2}} \\ \widehat{s}^2 &= \frac{s^2_{r_1}s^2_{r_2}}{s^2_{r_1} + s^2_{r_2}} \\ \widetilde{\mu} &= \frac{\mu_{**}\widehat{s}^2 + \widehat{m}\sigma^2}{\widehat{s}^2 + \sigma^2} \\ \widetilde{\sigma}^2 &= \frac{\widehat{s}^2\sigma^2}{\widehat{s}^2 + \sigma^2} \end{aligned}$$

Again note that this can be thought of as a sequential Normal update where we update s_1 using s_2, then μ_{**} using both. We then use an initial value for x to

generate r_1 and r_2, both from the same discrete probability distribution $w_1^{**}, \ldots, w_{10}^{**}$ where

$$w_j^{**} = \frac{w_j f(x; m_j + log(2), s_j^2)}{\sum_{j=1}^{10} w_j f(x; m_j + log(2), s_j^2)}$$

is the probability of obtaining a value $r_i = j$ for $i = 1, 2$. These values of r_1 and r_2 are then used to generate x from $p(x \mid y, r_1, r_2)$ in the same way outlined in section 4.5.1. Once we have obtained a sample using this improved auxiliary mixture sampling method, we can compare the density produced once again with the true density and the density resulting from using `LinBUGS`. This is shown in Figure 4.9 which is a vast improvement on Figure 4.7 where the approximation was produced using the original method.

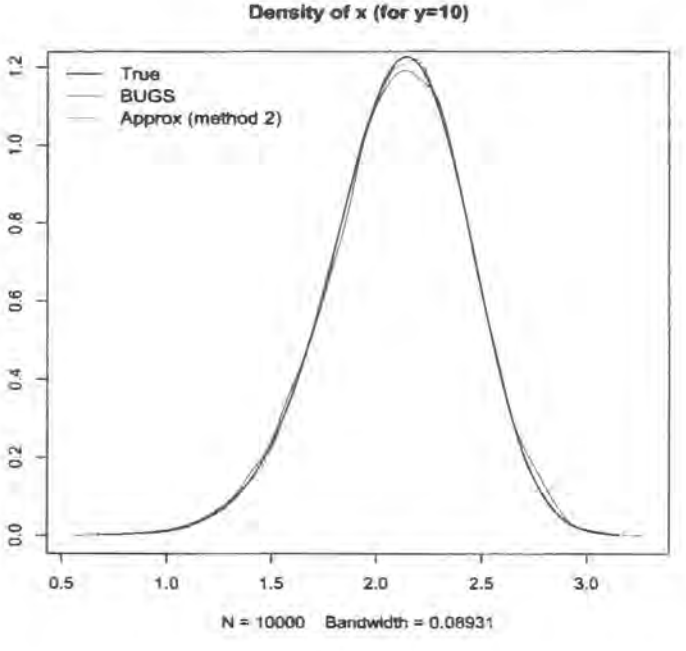


Figure 4.9: Improved auxiliary mixture approximation for large y

Note that we could take this further and include more mixture distributions to make the method possible for any value of y but this would increase the computational cost and therefore may not be any improvement on using another sampling scheme such as Metropolis-Hastings to solve our problem. Recall that our overall goal is to make the multivariate distribution given in equation (4.1) take a standard

distributional form. We now generalise our method to a multivariate case and illustrate that our goal is possible using a simplified version of (4.1) for one time point. We apply the NHS Direct data analysed in chapter 3 to do this for which a total of two mixture distributions is sufficient.

4.5.2 Multivariate case

Suppose that we now consider the following p-dimensional case

$$\begin{aligned} \mathbf{X} &\sim N_p(\boldsymbol{\mu}, \Sigma) \\ Y_i \mid X_i &\sim Poisson(E_i e^{X_i}) \end{aligned}$$

where $\boldsymbol{\mu}$, Σ, $\mathbf{y}$ and $E_1, \ldots, E_p$ are known and we want to use Gibbs sampling to generate from

$$p(\mathbf{x} \mid \mathbf{y}) \propto exp\Big\{ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\Big\} \prod_{i=1}^{p} exp\Big\{ x_i y_i - e^{logE_i + x_i}\Big\}.$$

We begin by noting that

$$\begin{aligned} exp\Big\{ y_i x_i - e^{logE_i+x_i}\Big\} \propto\ & e^{(y_i-2)x_i} exp\Big\{ log(E_i/2) + x_i - e^{log(E_i/2)+x_i}\Big\} \\ & \times exp\Big\{ log(E_i/2) + x_i - e^{log(E_i/2)+x_i}\Big\} \end{aligned}$$

which leads to

$$p(\mathbf{x} \mid \mathbf{y}) \propto exp\Big\{ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_*)'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_*)\Big\} \prod_{i=1}^{p} \varphi_{1\cdot i}\Big(log(E_i/2)+x_i\Big)\varphi_{2\cdot i}\Big(log(E_i/2)+x_i\Big) \tag{4.25}$$

where $\boldsymbol{\mu}_* = \boldsymbol{\mu} + \Sigma(\mathbf{y}-\mathbf{2})$ and $\varphi_{j\cdot i}(\cdot)$ for $i = 1, \ldots, p$ and $j = 1, 2$ is as defined in (4.23). Now we have

$$\begin{aligned} p(\mathbf{x} \mid \mathbf{y}) \overset{\propto}{\sim}\ & exp\Big\{ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_*)'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_*)\Big\} \\ & \times \prod_{i=1}^{p} \Big\{ \sum_{r_{1\cdot i}=1}^{10} w_{r_{1\cdot i}} f(x_i; m^*_{r_{1\cdot i}}, s^2_{r_{1\cdot i}}) \sum_{r_{2\cdot i}=1}^{10} w_{r_{2\cdot i}} f(x_i; m^*_{r_{2\cdot i}}, s^2_{r_{2\cdot i}})\Big\} \end{aligned}$$

where $m^*_{r_{j\cdot i}} = m_{r_{j\cdot i}} - log(E_i/2)$ for $j = 1, 2$ and $i = 1, \ldots, p$ and $f(\cdot)$ is as defined in section 4.5.1. If we let $\mathbf{r}_j = (r_{j\cdot 1}, \ldots, r_{j\cdot 12})$ for $j = 1, 2$ it follows that

$$\begin{aligned} p(\mathbf{x} \mid \mathbf{y}, \mathbf{r}_1, \mathbf{r}_2) &\overset{\propto}{\sim} exp\Big\{ -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_*)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_*) \Big\} \\ &\quad \times \prod_{i=1}^{p} exp\Big\{ -\frac{1}{2s^2_{r_{1\cdot i}}} (x_i - m^*_{r_{1\cdot i}})^2 \Big\} exp\Big\{ -\frac{1}{2s^2_{r_{2\cdot i}}} (x_i - m^*_{r_{2\cdot i}})^2 \Big\} \\ &\propto exp\Big\{ -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_*)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_*) \Big\} \prod_{i=1}^{p} exp\Big\{ -\frac{1}{2\widehat{s}^2_i} (x_i - \widehat{m}_i)^2 \Big\} \\ &\propto exp\Big\{ -\frac{1}{2}(\mathbf{x} - \widehat{\boldsymbol{\mu}})^T \widehat{\Sigma}^{-1} (\mathbf{x} - \widehat{\boldsymbol{\mu}}) \Big\} \end{aligned}$$

where

$$\begin{aligned} \widehat{m}_i &= \frac{m^*_{r_{1\cdot i}} s^2_{r_{2\cdot i}} + m^*_{r_{2\cdot i}} s^2_{r_{1\cdot i}}}{s^2_{r_{1\cdot i}} + s^2_{r_{2\cdot i}}} \\ \widehat{\mathbf{m}}^T &= (\widehat{m}_1, \ldots, \widehat{m}_{12}) \\ \widehat{s}^2_i &= \frac{s^2_{r_{1\cdot i}} s^2_{r_{2\cdot i}}}{s^2_{r_{1\cdot i}} + s^2_{r_{2\cdot i}}} \\ S &= diag\Big(\frac{1}{\widehat{s}^2_1}, \ldots, \frac{1}{\widehat{s}^2_{12}}\Big) \\ \widehat{\boldsymbol{\mu}} &= \Big[(\boldsymbol{\mu}_*^T \Sigma^{-1} + \widehat{\mathbf{m}}^T S) \widehat{\Sigma} \Big]^T \\ \widehat{\Sigma} &= \Big(\Sigma^{-1} + S \Big)^{-1} \end{aligned}$$

We then use initial value $\mathbf{x}$ to generate each $r_{j\cdot i} \mid x_i$ for $i = 1, \ldots, p$ and $j = 1, 2$ from discrete probability distribution $w^*_{i\cdot 1}, \ldots, w^*_{i\cdot 10}$ where

$$w^*_{i\cdot k} = \frac{w_k f(x_i; m_k - log(E_i/2), s^2_k)}{\sum_{k=1}^{10} w_k f(x_i; m_k - log(E_i/2), s^2_k)}$$

is the probability of obtaining a value $r_{j\cdot i} = k$. These values of $\mathbf{r}_1$ and $\mathbf{r}_2$ are then used to generate $\mathbf{x}$ from $p(\mathbf{x} \mid \mathbf{y}, \mathbf{r}_1, \mathbf{r}_2)$ in the same way outlined in section 4.5.1.

Example

We now return our attention to the model of Mugglin *et al.* [36] described in chapter 3 and illustrate results for the particular problem outlined in equation (4.1). Restricting attention to $t = 1$, our problem is simply the multivariate example of this section where $p = 12$. We first use our known values for $\boldsymbol{\mu}$, Σ, $\mathbf{y}$, $E_1, \ldots, E_{12}$ and chosen initial $\mathbf{x}$ to obtain a sample of size 10000 using the method described in

this section. We then use the same $\boldsymbol{\mu}$, Σ, $\mathbf{y}$, $E_1, \ldots, E_{12}$ and initial $\mathbf{x}$ and obtain an equivalent sample using `LinBUGS`. Both sets of densities obtained are compared in Figure 4.10 and the relative entropy for each pair of densities is shown below each plot. The densities look to be very close and this is confirmed by the small relative entropys between them.

We also want to verify that the joint distributional structure of the posterior distributions are similar whether samples are obtained using our auxiliary mixture approximation method or using LinBUGS. Using the `R` package `CODA` we find the cross correlations between each of the pairs of variables for both of the posterior samples and then find the difference between the two. Table 4.3 shows these differences for the first 6 variables. We can see that they are all relatively small and continuing for all 12 variables produces a maximum difference of 0.086 which is still sufficiently small to deduce that the joint distributional structures are similar.

Table 4.3: Differences in cross-correlations obtained using the two approaches

	x_1	x_2	x_3	x_4	x_5	x_6	$\cdots$
x_1	0.000	0.023	0.030	0.015	0.004	0.012	$\cdots$
x_2		0.000	0.012	0.001	0.016	0.037	$\cdots$
x_3			0.000	0.024	0.014	0.019	$\cdots$
x_4				0.000	0.022	0.007	$\cdots$
x_5					0.000	0.002	$\cdots$
x_6						0.000	$\cdots$
$\vdots$							$\ddots$

We can therefore conclude that the auxiliary mixture approximation method works as well as `LinBUGS` for the multivariate case.

4.6 Properties of the multivariate auxiliary approximation method

In this section we explore further the statistical properties of the multivariate auxiliary mixture approximation method. Ideally we would like to compare the samples

obtained using this method with the true posterior distribution as we did for the univariate case (for example in Figure 4.9) but computation of the true posterior in this case is very difficult. Instead, we use many simulated examples (described further in section 4.6.1) and present three methods of analysis which provide valuable insight into the properties and performance of our method.

4.6.1 Simulated examples

Suppose that we continue using 12 dimensions as we did in the example of section 4.5.2 so we have

$$\begin{aligned} \mathbf{X} &\sim N_{12}(\boldsymbol{\mu}, \Sigma) \\ Y_i \mid X_i &\sim Poisson(E_i e^{X_i}) \end{aligned}$$

where Σ is kept the same as in the example and $E_1, \ldots, E_{12} = 1$. We consider this example for three different data sizes and obtain a sample of size 10000 from the posterior for each. The three different cases we consider are as follows

1. where the data have small counts. More specifically, we let $\mu_i = log(1)$ for $i = 1, \ldots, 12$ to allow each $E[Y_i \mid X_i] = 1$.

2. where the data have slightly larger counts. This time we let $\mu_i = log(4)$ for $i = 1, \ldots, 12$ to allow each $E[Y_i \mid X_i] = 4$.

3. where the data have large counts. Here we let $\mu_i = log(30)$ for $i = 1, \ldots, 12$ to allow each $E[Y_i \mid X_i] = 30$.

For each case we simulate an initial value $\mathbf{x} = \mathbf{x_0}$ from the true distribution $N_{12}(\boldsymbol{\mu}, \Sigma)$ then use this value to simulate $\mathbf{y}$ from the distribution $Poisson(E_i e^{X_i})$. We then use the method of section 4.5.2 to obtain a sample $\mathbf{x}_1, \ldots, \mathbf{x}_{10000}$ from the posterior. This is then repeated 100 times giving us 100 sets of 10000 samples for each of the three cases.

4.6.2 Checking that posterior contains true value

We begin by checking that our posterior sample does actually contain our 'true' value of $\mathbf{x}$. We do this univariately to start with and consider where x_{0i} lies in

$x_{1i}, \ldots, x_{10000i}$. We find what proportion of the sample is $\leq x_{0i}$ and call this value p_i. For each of the three cases we obtain 100 values of p_i. Since the distribution of P_i tends to Uniform(0,1) as the sample size tends to infinity, we use a Chi-squared test to test each of the three groups of p_is for Uniformity. Before we do this however, we first present a proof of the claim that the distribution of P_i is Uniform.

Consider the following algorithm:

1. Generate $\mathbf{x}_0$ from the distribution of $\mathbf{X}$
2. Generate $\mathbf{y}_0$ from the distribution of $\mathbf{Y} \mid \mathbf{X} = \mathbf{x}_0$
3. Generate a sample $\mathbf{x}_1^*, \ldots, \mathbf{x}_N^*$ from the distribution $\mathbf{X}^* \mid \mathbf{Y}$ where $\mathbf{X}^* \mid \mathbf{Y} \equiv \mathbf{X} \mid \mathbf{Y}$ if the algorithm works
4. Find what proportion of $x_{1i}^*, \ldots, x_{Ni}^*$ are $\leq x_{0i}$ and call this $\widehat{p}_i$

Claim

If the above algorithm works then $\widehat{P}_i \xrightarrow{d} Unif(0,1)$ as $N \to \infty$

Proof

We know that

$$\widehat{P}_i \to F_{X_i^* \mid \mathbf{Y}}(x_{0i} \mid \mathbf{y}_0) \text{ as } N \to \infty.$$

Suppose that the algroithm works. Then

$$F_{X_i^* \mid \mathbf{Y}} \equiv F_{X_i \mid \mathbf{Y}}$$

so it follows that

$$\widehat{P}_i \approx P_i = F_{X_i \mid \mathbf{Y}}(X_{0i} \mid \mathbf{y}_0).$$

Since $\mathbf{x}_0, \mathbf{y}$ are sampled from the joint distribution of $(\mathbf{X}, \mathbf{Y})$ it means that

$$\mathbf{X}_0 \mid \mathbf{Y} = \mathbf{y}_0 \overset{d}{=} \mathbf{X} \mid \mathbf{Y} = \mathbf{y}_0.$$

We know that for any continuous univariate distribution we have

$$F_W(W) \sim Uniform(0,1),$$

so this implies that

$$P_i \mid \mathbf{Y} = \mathbf{y}_0 \sim Uniform(0,1)$$

for all $\mathbf{y}_0$, which means that

$$P_i \sim Uniform(0,1).$$

Note that if $\mathbf{X}^* \mid \mathbf{Y} \neq \mathbf{X} \mid \mathbf{Y}$, then $p_i = F_{X_i^* \mid \mathbf{Y}}(x_{0i} \mid \mathbf{y})$ will not have a uniform distribution for all $\mathbf{y}_0$, unless for some reason $F_{X_i^* \mid \mathbf{Y}} \equiv F_{X_i \mid \mathbf{Y}}$.

Results

For each of the three data cases and for $i = 1, \ldots, 12$ we have 100 values of p_i. We then use the Chi-squared test on each of these 100 values to test for uniformity. Table 4.4 shows the p-values resulting from the test for each data case and each value of i.

i	p when $\mu_i = log(1)$	p when $\mu_i = log(4)$	p when $\mu_i = log(30)$
1	0.15	0.80	0.15
2	0.44	0.32	0.26
3	0.83	0.21	0.19
4	0.20	0.85	0.12
5	0.87	0.14	0.09
6	0.42	0.32	0.82
7	0.09	0.92	0.03
8	0.47	0.44	0.99
9	0.24	0.60	0.85
10	0.78	0.53	0.62
11	0.62	0.72	0.15
12	0.83	0.91	0.74

Table 4.4: p-values resulting from the Chi-squared test for uniformity

In general these p-values are high enough not to reject the claim that the values of p_i we have could be uniform distributed.

4.6.3 Mahalanobis distance

Mahalanobis distance is a useful way of determining similarity between two groups of values. It differs from Euclidean distance in that it takes correlations into account. Suppose we wanted to find the similarity between a set of values $\mathbf{x}$ and another set of values with mean $\mathbf{m}$ and covariance matrix C then the Mahalanobis distance would be

$$d = (\mathbf{x} - \mathbf{m})' C^{-1} (\mathbf{x} - \mathbf{m}).$$

To see how Mahalanobis distance works we use our true set of values $\mathbf{x}_0$ for $\mathbf{x}$ along with prior mean μ and covariance matrix Σ for $\mathbf{m}$ and C. For any multivariate Normal distribution X with given mean vector and variance matrix, the Mahalanobis distance has exactly a Chi-squared distribution with the degrees of freedom equal to the rank of the variance matrix. We found this distance for each of the 100 $\mathbf{x}_0$ values we have for each of the three data cases. The summaries of these distances are shown in Table 4.5 separately for each data case and each one could be said to follow the χ^2_{12} distribution. One indication of this is that each of the summaries are consistent with the χ^2_{12} distribution which has a mean of 12, a standard deviation of 4.90 and a median of approximately 11.33.

	d when $\mu_i = log(1)$	d when $\mu_i = log(4)$	d when $\mu_i = log(30)$
Min.	5.14	4.84	4.72
1st Qu.	9.51	9.44	9.38
Median	10.95	10.80	11.21
Mean	12.24	10.74	11.54
3rd Qu.	13.26	12.91	13.80
Max.	25.85	15.47	17.75

Table 4.5: Mahalanobis distance between true values and prior

We now use Mahalanobis distance to consider the similarity between our true value $\mathbf{x}_0$ and the sample $\mathbf{x}_1, \ldots, \mathbf{x}_{10000}$ for which we have mean $\bar{\mathbf{x}}$ and covariance matrix C. We obtain 100 values of this distance d_0 for each of the three data cases. Table 4.6 shows summaries of these 100 values for each of the data cases separately.

	d_0 when $\mu_i = log(1)$	d_0 when $\mu_i = log(4)$	d_0 when $\mu_i = log(30)$
Min.	2.09	1.90	953.2
1st Qu.	8.14	9.19	3073.0
Median	11.16	11.79	4433.0
Mean	11.89	18.17	5245.0
3rd Qu.	14.42	16.38	6992.0
Max.	27.47	370.20	14030.0

Table 4.6: Mahalanobis distance between true values and samples

The distances in the first column of Table 4.6 seem fine and we can assume that the posterior contains the true value. Although the second column is fine on the whole, it does include some large distances which indicate that there may be some break down in our method for that data case. By the time we get to the third column we see evidence of a definite problem with our method when the data counts are large.

We can also use Mahalanobis distance to find the similarity between each sample $\mathbf{x}_j$ and another set of values which have the sample mean $\bar{\mathbf{x}}$ and sample covariance matrix C. Suppose we call these distances d_j where $j = 1, \ldots, 10000$. We can then find where the distance d_0 lies in $d_1, \ldots, d_{10000}$ for each of the 100 simulations and each of the three data cases. The summaries of these quantiles are given in Table 4.7.

	$\mu_i = log(1)$	$\mu_i = log(4)$	$\mu_i = log(30)$
Min.	0.001	0.001	1
1st Qu.	0.223	0.315	1
Median	0.482	0.538	1
Mean	0.489	0.534	1
3rd Qu.	0.729	0.827	1
Max.	0.994	1.000	1

Table 4.7: Quantiles of d_0 in $d_1, \ldots, d_{10000}$

The median and mean tell us that the quantiles for the first two data cases are most around 0.5 , which is what we would expect. However for the large data case

we see that the true value is always larger than the posterior, which again indicates there is a problem.

4.6.4 Variance decomposition

In this section we consider a well known property of variance to give us further insight into how our method is working. Since we know that

$$Var[X] = Var\big[E[X \mid Y]\big] + E\big[Var[X \mid Y]\big], \tag{4.26}$$

it should be the case that, for each of the three data cases, the original Σ should be approximately equal to the expectation of the variance of the samples added to the variance of the expectation of the samples. For each of the three data cases we do the following

1. Find $E[X \mid Y]$: for each of the 100 simulations we find the mean of the sample values $\mathbf{x}_1, \ldots, \mathbf{x}_{10000}$ as follows

$$m_i = \sum_{j=1}^{10000} x_{ji} \text{ for } i = 1, \ldots, 12$$

   so we have 12 mean vectors $\mathbf{m}_i$ each of length 100.

2. Find $Var\big[E[X \mid Y]\big]$: we find the covariance matrix for the 12 mean vectors.

3. Find $Var[X \mid Y]$: for each of the 100 simulations we find the covariance matrix for the mean vectors $\widehat{\mathbf{x}}_1, \ldots, \widehat{\mathbf{x}}_{12}$ where

$$\widehat{\mathbf{x}}_i = (x_{1i}, \ldots, x_{10000i}) \text{ for } i = 1, \ldots, 12$$

   so we have 100 12 by 12 covariance matrices.

4. Find $E\big[Var[X \mid Y]\big]$: we average the 100 covariance matrices to obtain one 12 by 12 matrix.

5. Add $E\big[Var[X \mid Y]\big]$ to $Var\big[E[X \mid Y]\big]$ and compare with $Var[X] = \Sigma$

For the first case, where the data have small counts, we find that there is little difference between the two matrices obtained at step 5. Comparing element by

element of the matrices we find the largest difference between the elements to be 0.026 and the mean difference to be 0.004 which is very small. For the second data case, the two matrices could still be thought of as similar. The mean difference between comparible elements is 0.124 and the maximum difference is 1.350 which is still relatively small. However, by the time we reach the third data case the matrices are quite different. The mean difference between individual elements is 14.828 and the maximum reaches 90.511.

We then convert the prior covariance matrix Σ into the correlation matrix by scaling by standard deviation and use the same rescaling on the right hand side of equation (4.26). For each of the three data cases we let

$$\begin{aligned} V &= \frac{\Sigma}{\text{scale}} \\ V_1 &= \frac{Var[E[X \mid Y]]}{\text{scale}} \\ V_2 &= \frac{E[Var[X \mid Y]]}{\text{scale}} \end{aligned}$$

and then look at the element by element differences between V and $(V_1 + V_2)$. Note here that it is possible for matrices V_1 and V_2 to have values greater than 1 since we are using the same scaling that we use to convert the Σ into a correlation matrix. However any such values would definitely indicate a break down of the method. Tables 4.8 to 4.11 show matrices V, V_1, V_2 and $V - (V_1 + V_2)$ for the first data case.

Table 4.11 has the maximum absolute value of 0.06 meaning that our method works well for the first data case. However for the second data case, the maximum absolute difference between V and $(V_1 + V_2)$ is 3.55 which indicates a break down of our method. Furthermore, the third data case has maximum absolute difference of 232.471 which means that our method is definitely failing.

4.6.5 The role of the y values

In order to investigate this problem further we look at the values of **y** produced during our simulations. Table 4.12 shows a summary of these broken down by data case.

In section 4.5.1 starting from page 87 we consider how well the univariate version

Table 4.8: V for data case when $\mu_i = 1$

1.00	0.04	0.04	0.16	0.01	0.02	0.01	0.01	0.04	0.03	0.01	0.01
0.04	1.00	0.26	0.25	0.07	0.26	0.10	0.07	0.25	0.09	0.08	0.10
0.04	0.26	1.00	0.26	0.23	0.27	0.09	0.05	0.14	0.24	0.10	0.25
0.16	0.25	0.26	1.00	0.08	0.14	0.07	0.05	0.22	0.21	0.05	0.08
0.01	0.07	0.23	0.08	1.00	0.09	0.03	0.02	0.05	0.20	0.05	0.21
0.02	0.26	0.27	0.14	0.09	1.00	0.25	0.12	0.27	0.08	0.25	0.25
0.01	0.10	0.09	0.07	0.03	0.25	1.00	0.23	0.24	0.03	0.24	0.09
0.01	0.07	0.05	0.05	0.02	0.12	0.23	1.00	0.21	0.02	0.21	0.06
0.04	0.25	0.14	0.22	0.05	0.27	0.24	0.21	1.00	0.06	0.12	0.09
0.03	0.09	0.24	0.21	0.20	0.08	0.03	0.02	0.06	1.00	0.03	0.09
0.01	0.08	0.10	0.05	0.05	0.25	0.24	0.21	0.12	0.03	1.00	0.21
0.01	0.10	0.25	0.08	0.21	0.25	0.09	0.06	0.09	0.09	0.21	1.00

Table 4.9: V_1 for data case when $\mu_i = 1$

0.09	0.02	0.00	0.04	0.01	0.01	0.01	0.02	0.02	0.03	0.01	0.03
0.02	0.19	0.11	0.11	0.03	0.11	0.05	0.06	0.12	0.06	0.06	0.04
0.00	0.11	0.24	0.13	0.09	0.13	0.06	0.07	0.10	0.11	0.08	0.13
0.04	0.11	0.13	0.22	0.06	0.07	0.01	0.05	0.11	0.10	0.02	0.04
0.01	0.03	0.09	0.06	0.11	0.05	0.04	0.03	0.04	0.07	0.04	0.08
0.01	0.11	0.13	0.07	0.05	0.22	0.16	0.08	0.16	0.08	0.12	0.12
0.01	0.05	0.06	0.01	0.04	0.16	0.41	0.17	0.17	0.03	0.15	0.05
0.02	0.06	0.07	0.05	0.03	0.08	0.17	0.36	0.17	0.04	0.12	0.04
0.02	0.12	0.10	0.11	0.04	0.16	0.17	0.17	0.39	0.07	0.10	0.08
0.03	0.06	0.11	0.10	0.07	0.08	0.03	0.04	0.07	0.18	0.02	0.04
0.01	0.06	0.08	0.02	0.04	0.12	0.15	0.12	0.10	0.02	0.31	0.13
0.03	0.04	0.13	0.04	0.08	0.12	0.05	0.04	0.08	0.04	0.13	0.34

of our method works for large y. In particular we consider when $y = 10$ and conclude that it works well in that case (see Figure 4.9 for example). We also noted that it would be possible to make the method work for any value of y by adding more mixture distributions but that doing so would increase the computational cost. This

Table 4.10: V_2 for data case when $\mu_i = 1$

0.88	0.02	0.02	0.11	0.00	0.01	0.00	0.00	0.01	0.02	0.00	0.00
0.02	0.80	0.15	0.14	0.03	0.15	0.04	0.02	0.12	0.04	0.03	0.04
0.02	0.15	0.78	0.15	0.15	0.16	0.03	0.01	0.05	0.14	0.04	0.13
0.11	0.14	0.15	0.77	0.04	0.06	0.02	0.02	0.11	0.12	0.02	0.03
0.00	0.03	0.15	0.04	0.87	0.05	0.01	0.01	0.01	0.13	0.02	0.12
0.01	0.15	0.16	0.06	0.05	0.81	0.12	0.04	0.13	0.03	0.13	0.13
0.00	0.04	0.03	0.02	0.01	0.12	0.66	0.10	0.10	0.01	0.10	0.03
0.00	0.02	0.01	0.02	0.01	0.04	0.10	0.71	0.09	0.01	0.10	0.02
0.01	0.12	0.05	0.11	0.01	0.13	0.10	0.09	0.64	0.02	0.04	0.03
0.02	0.04	0.14	0.12	0.13	0.03	0.01	0.01	0.02	0.78	0.01	0.04
0.00	0.03	0.04	0.02	0.02	0.13	0.10	0.10	0.04	0.01	0.69	0.09
0.00	0.04	0.13	0.03	0.12	0.13	0.03	0.02	0.03	0.04	0.09	0.67

Table 4.11: $V - (V_1 + V_2)$ for data case when $\mu_i = 1$

0.03	0.00	0.02	0.01	0.00	0.00	-0.01	-0.02	0.00	-0.02	0.00	-0.02
0.00	0.01	-0.01	-0.01	0.00	0.00	0.01	-0.01	0.01	-0.01	-0.01	0.02
0.02	-0.01	-0.03	-0.02	0.00	-0.02	0.00	-0.03	-0.01	-0.02	-0.02	-0.01
0.01	-0.01	-0.02	0.01	-0.02	0.01	0.04	-0.01	0.01	-0.02	0.02	0.01
0.00	0.00	0.00	-0.02	0.02	0.00	-0.01	-0.02	0.00	0.00	-0.01	0.01
0.00	0.00	-0.02	0.01	0.00	-0.03	-0.03	0.00	-0.02	-0.03	0.00	0.01
-0.01	0.01	0.00	0.04	-0.01	-0.03	-0.06	-0.04	-0.02	-0.01	-0.01	0.02
-0.02	-0.01	-0.03	-0.01	-0.02	0.00	-0.04	-0.07	-0.05	-0.03	-0.01	0.01
0.00	0.01	-0.01	0.01	0.00	-0.02	-0.02	-0.05	-0.03	-0.03	-0.01	-0.02
-0.02	-0.01	-0.02	-0.02	0.00	-0.03	-0.01	-0.03	-0.03	0.04	0.00	0.01
0.00	-0.01	-0.02	0.02	-0.01	0.00	-0.01	-0.01	-0.01	0.00	0.01	-0.01
-0.02	0.02	-0.01	0.01	0.01	0.01	0.02	0.01	-0.02	0.01	-0.01	-0.01

is the same for the multivariate case which we are considering here. From Table 4.12 we can see that the y values in the first data case all fall within the boundaries that we know the method works for. For the second data case we can see that they mostly fall within these boundaries but do contain at least one very large value for

	y_i when $E[Y_i \mid X_i] = 1$	y_i when $E[Y_i \mid X_i] = 4$	y_i when $E[Y_i \mid X_i] = 30$
Min.	0.0	0.0	4.0
1st Qu.	0.0	2.0	21.0
Median	1.0	4.0	31.0
Mean	1.2	4.6	34.8
3rd Qu.	2.0	6.0	43.0
Max.	10.0	31.0	183.0

Table 4.12: Summary of y values

which our method may not work. By the time we reach the third case we can see that the majority of the y values simulated don't fit within these boundaries and therefore are likely to be the cause of the break down of our method. To fix this we would alter equation (4.25) to include more φs for example,

$$p(\mathbf{x} \mid \mathbf{y}) \propto exp\Big\{ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_*)'\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_*)\Big\} \prod_{i=1}^{p} \varphi_{1\cdot i}\Big(log(E_i/4)+x_i\Big) \times \varphi_{2\cdot i}\Big(log(E_i/4)+x_i\Big)\varphi_{3\cdot i}\Big(log(E_i/4)+x_i\Big)\varphi_{4\cdot i}\Big(log(E_i/4)+x_i\Big).$$

However, as we have already stated, this would increase the computational cost of the method.

We noted in section 4.2 that one of the most well known methods for addressing the problem in this chapter is presented in Rue *et al.* [45] which uses a Gaussian approximation to $p(x \mid y)$. Recall that when the observed counts are small, the Poisson term is no longer approximated well by a Gaussian term so their method runs into problems. In contrast, our method does work well when the counts are small and can be improved to also work well for large counts.

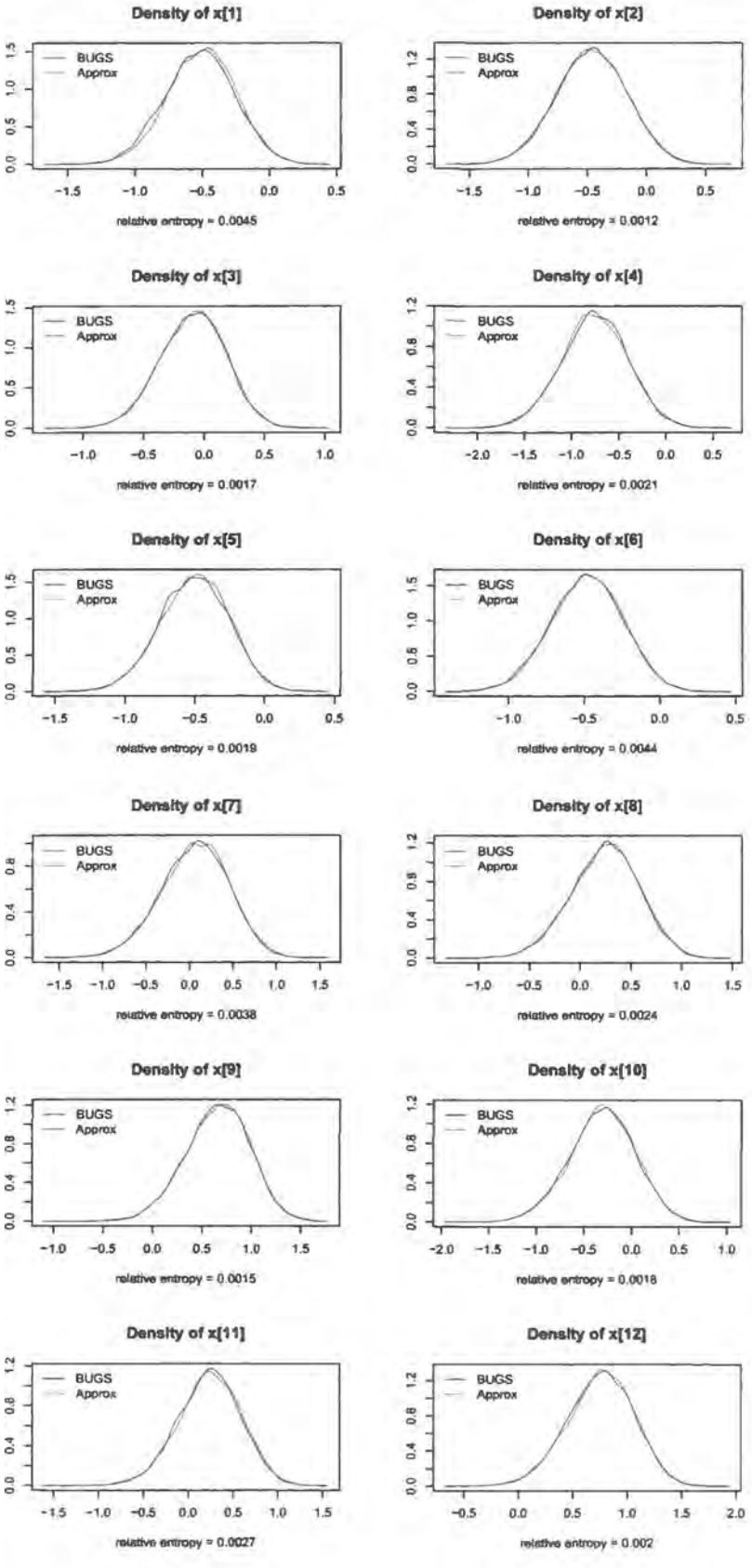


Figure 4.10: Accuracy of the approximation for the multivariate case

Chapter 5

Prior sensitivity analysis of MCMC output

5.1 Background

The introduction of Markov chain Monte Carlo (MCMC) computational methods has had a dramatic effect on assessing prior sensitivity. MCMC allows modelling of such complexity that inputs such as priors can only be elicited in a very casual way thereby increasing the need for sensitivity considerations. (Berger *et al.* [3]).

Recall from section 1.2.2 that there has been much work focussing on local sensitivity where small changes to the prior are studied. As we mentioned, one such example is McCulloch [34] who develops a general method for assessing the influence of model assumptions in a Bayesian analysis. He looks at the effect of changing the hyperparameter away from the initial choice and uses relative entropy to measure the difference between the posteriors resulting from different choices of hyperparameter. However, this approach requires us to know the resulting posterior distributions, which is not always possible. In the case where MCMC methods are used, we do not know the posterior distribution exactly but instead have only a sample from it. Clarke and Gustafson [7], whose work extends the idea of McCulloch [34], suggest how this method could be applied in the case where MCMC methods are used but they do not pursue this. In considering this idea further we discover a drawback of this approach. It doesn't allow us to see exactly which parts of the posterior are

affected by the changes to the prior, which would be both interesting and useful to know. We explain this further in section 5.1.1.

As well as not knowing the exact posterior distribution, another problem encountered when using MCMC is computational time. It can be very time consuming to run an initial model using MCMC, obtain a sample, change the prior parameter values and then re-run it to obtain another sample. Smith and Gelfand [49] present one solution to this in that once they obtain a sample from one posterior, they use importance sampling to obtain a sample from the posterior resulting from a 'close' prior.

In this chapter we develop a practical way to see how sensitive each parameter in the model is to specified prior changes. Using just one MCMC run we obtain a sample from the posterior distribution of each parameter in the model and estimate the densities using kernel density estimation. Using the same sample and by incorporating the idea of importance sampling we also estimate the densities of marginal posterior distributions resulting from a different prior. We then quantify sensitivity for each parameter by finding the relative entropy between the corresponding marginal density from the first set of densities we estimated and that from the second. This is something that doesn't seem to have been covered in previous Bayesian prior sensitivity work.

In the remainder of this section we look in more depth at a sensitivity approach currently in the literature and also explore relative entropy in more detail. In section 5.2 we develop the new marginal sensitivity method described above and in section 5.3 consider how well it works. In section 5.4 we see how changing the metric from relative entropy to Kolmogorov distance affects the results and in section 5.5 we apply the method to the model and data of chapter 3.

5.1.1 A current sensitivity method

Clarke and Gustafson [7] propose a method to quantify the local sensitivity of the posterior to simultaneous changes in the three inputs - the prior, the model and the data. For simplicity they restrict attention to conjugate priors and describe their method in the context of an example where $X \sim Gamma(\lambda, \theta/\lambda)$,

$\theta \sim InverseGamma(\alpha_1, \alpha_2)$ and $\theta \mid X \sim InverseGamma(\alpha_1 + n\lambda, \alpha_2 + n\lambda\bar{x})$. They define the baseline set of inputs $\omega = (\alpha, \lambda, x)$ and a nearby set of inputs $\tilde{\omega} = (\tilde{\alpha}, \tilde{\lambda}, \tilde{x})$. The discrepancy between the two posteriors arising from the two sets of inputs is measured using relative entropy

$$d_{ps}(\omega, \tilde{\omega}) = D\big(p(\theta \mid x; \omega) \big\| p(\theta \mid x; \tilde{\omega})\big) \tag{5.1}$$

where $D\big(p(x)\big\|q(x)\big) = \int p(x) log(p(x)/q(x)) dx$ for densities p and q. The second order Taylor expansions of (5.1) about ω is approximately

$$d^*_{PS}(\omega, \tilde{\omega}) = \frac{1}{2}(\tilde{\omega} - \omega)^T A_{PS}(\omega)(\tilde{\omega} - \omega) \tag{5.2}$$

where $A_{PS}(\omega)$ is the second derivative of $d_{PS}(\omega, \tilde{\omega})$ with respect to $\tilde{\omega}$ evaluated at $\tilde{\omega} = \omega$. This $A_{PS}(\omega)$ can be thought of as the Fisher information matrix for the $InverseGamma(\alpha_1 + n\lambda, \alpha_2 + n\lambda\bar{x})$ family of distributions.

For situations where we do not know the posterior distribution exactly but instead have only an MCMC sample from it, the Fisher information matrix $A_{PS}(\omega)$ can be expressed as

$$\Big[A_{PS}(\omega)\Big]_{ij} = Cov\Big(\frac{\partial}{\partial\omega_i}[logp(\vartheta;\omega) + logL(\vartheta;\omega)], \frac{\partial}{\partial\omega_j}[logp(\vartheta;\omega) + logL(\vartheta;\omega)]\Big) \tag{5.3}$$

where $p(\vartheta;\omega)$ is the prior density and $L(\vartheta;\omega)$ is the likelihood. This can be estimated using the sample covariance.

Suppose now that we are only considering changes to the prior distribution so that ω comprises only prior parameters. In this case the likelihood does not depend on ω so $\frac{\partial}{\partial\omega_i}[logL(\vartheta;\omega)] = 0$ and equation (5.3) can be rewritten as

$$\Big[A_{PS}(\omega)\Big]_{ij} = Cov\Big(\frac{\partial}{\partial\omega_i} logp(\vartheta;\omega), \frac{\partial}{\partial\omega_j} logp(\vartheta;\omega)\Big)$$

To illustrate a drawback of this approach we consider a prior distribution comprising n parameters $p(\vartheta;\omega) = p(\vartheta_1;\omega_1)p(\vartheta_2;\omega_2)\cdots p(\vartheta_n;\omega_n)$ with

$$logp(\vartheta;\omega) = logp(\vartheta_1;\omega_1) + logp(\vartheta_2;\omega_2) + \ldots + logp(\vartheta_n;\omega_n)$$

It is easy to see from this that $\frac{\partial}{\partial\omega_i} logp(\vartheta;\omega) = \frac{\partial}{\partial\omega_i} logp(\vartheta_i;\omega_i)$ and therefore only involves the ϑ_i parameter. Similarly $\frac{\partial}{\partial\omega_j} logp(\vartheta;\omega)$ only involves the ϑ_j parameter

and it follows that for $i \neq j$

$$\Big[A_{PS}(\omega)\Big]_{ij} = Cov\Big(\frac{\partial}{\partial\omega_i} logp(\vartheta;\omega), \frac{\partial}{\partial\omega_j} logp(\vartheta;\omega)\Big) = 0$$

It is therefore impossible to use this method to see the effect that changing the prior of one parameter has on another.

5.1.2 Relative entropy

Relative entropy (or the *Kullback-Leibler divergence*) is defined by

$$D(P \parallel Q) = \int p(x) log\left(\frac{p(x)}{q(x)}\right) dx \tag{5.4}$$

and can be thought of as a general measure of the difference between two distributions P and Q with densities $p(x)$ and $q(x)$ respectively. It assumes values in $[0, \infty)$ and has some of the characteristics of a distance, namely $D(P \parallel Q) > 0$ and $D(P \parallel P) = 0$ but also has the property that $D(P \parallel Q) \neq D(Q \parallel P)$ unlike a distance. A better interpretation of $D(P \parallel Q)$ would possibly be the cost of using distribution Q when P is the correct one.

To get an idea of how relative entropy behaves we consider changes to the Normal and Gamma distributions. Suppose P_0 has a Normal distribution with mean μ_0 and variance $1/\tau_0$ and we change the parameters to produce another Normal distribution P_1 with mean μ_1 and variance $1/\tau_1$. The 'difference' between P_0 and P_1 is measured by

$$D\Big(P_0 \Big\| P_1\Big) = \frac{1}{2}\Big[\tau_1(\mu_1 - \mu_0)^2 + \frac{\tau_1}{\tau_0} + log\Big(\frac{\tau_0}{\tau_1}\Big) - 1\Big]. \tag{5.5}$$

Suppose P_0 has parameter values $\mu_0 = 0$ and $\tau_0 = 1$ and we change only one parameter at a time. Figure 5.1 (a) shows the relative entropy calculated for various values of μ_1 where τ_1 is kept at 1 and (c) shows the relative entropy calculated for various values of τ_1 where μ_1 is kept at 0. They also highlight some values of μ_1 and τ_1 that produce a relative entropy of sizes 0.5, 1 and 1.5. For example, the blue lines in (a) tell us that setting $\mu_1 \approx 1.8$ produces a relative entropy between P_0 and P_1 of 1.5. Figure 5.1 (b) and (d) show the original distribution P_0 in grey and three

different changed distributions P_1 in red, green and blue corresponding to changes of size 0.5, 1 and 1.5. Continuing the above example, the blue line in (b) corresponds to the distribution $P_1 \sim N(1.8, 1)$.

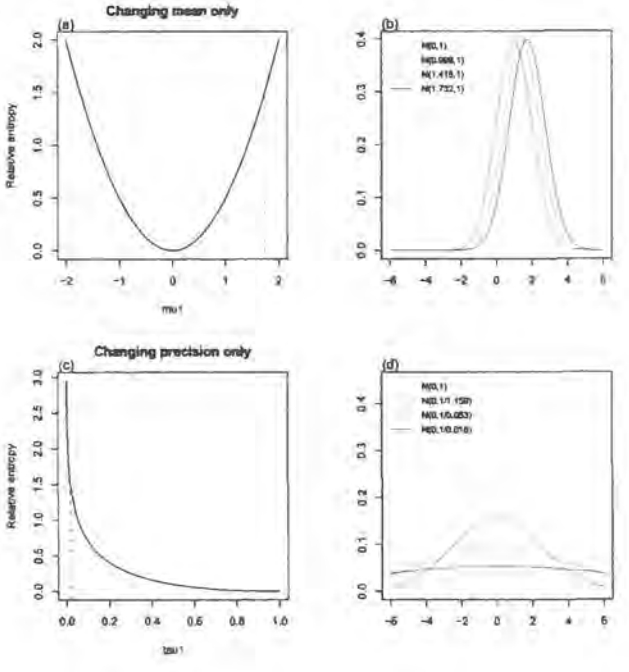


Figure 5.1: Changes to the Normal distribution measured by relative entropy

Suppose now that P_0 has a Gamma distribution with shape parameter a_0 and rate parameter b_0 and we change the parameters to produce another Gamma distribution P_1 with shape a_1 and rate b_1. One specification[1] for the 'difference' between P_0 and P_1 is given by

$$D\Big(P_0\|P_1\Big) = (a_0 - a_1)\Big(\psi(a_0) + log(b_0)\Big) + a_0\Big(\frac{b_1}{b_0} - 1\Big) + log\Big(\frac{b_0^{a_0}\Gamma(a_1)}{b_1^{a_1}\Gamma(a_0)}\Big) \qquad (5.6)$$

where $\psi(z) = \frac{d}{dz}\, log\,\Gamma(z)$ is the digamma function. Figure 5.2 gives us more of an idea of what this looks like for different values of a_1 and b_1 where $a_0 = b_0 = 1$.

[1] used in `http://en.wikipedia.org/wiki/Gamma_distribution`

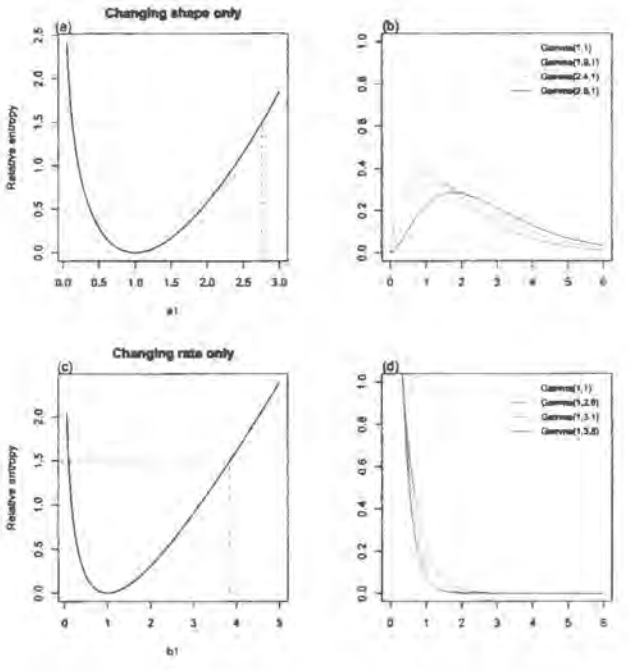


Figure 5.2: Changes to the Gamma distribution measured by relative entropy

5.2 Marginal sensitivity method

In this section we present a new method which enables us to specify a single change to the prior distribution and obtain a numerical value for how sensitive each of the model parameters are to this change. To begin with we assume that we have performed an MCMC simulation using a baseline prior distribution and have obtained a sample from the resulting posterior. We proceed by describing the method in the context of an example for which we need to specify some notation.

5.2.1 Notation

- The model has n prior parameters $\theta_1, \ldots, \theta_n$
- depending on hyperparameters $\omega_1, \ldots, \omega_n$ where $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots, \omega_n)$
- The baseline prior distribution is given by

$$p(\boldsymbol{\theta}; \boldsymbol{\omega}) = p(\theta_1; \omega_1)p(\theta_2; \omega_2) \cdots p(\theta_n; \omega_n)$$

- The baseline marginal posterior distributions are given by

$$p(\theta_j \mid \mathbf{x}, \boldsymbol{\omega}) \quad \text{for } j = 1, \ldots, n.$$

- from which we have MCMC sample

$$\theta_j^{[1]}, \ldots, \theta_j^{[N]} \quad \text{for } j = 1, \ldots, n.$$

Suppose for this example that we make changes to the prior of θ_1 only such that ω_1 becomes $\widetilde{\omega}_1$ but all other hyperparameters remain the same, i.e. $\widetilde{\boldsymbol{\omega}} = (\widetilde{\omega}_1, \omega_2, \ldots, \omega_n)$.

- The changed prior distribution is given by

$$p(\boldsymbol{\theta}; \widetilde{\boldsymbol{\omega}}) = p(\theta_1; \widetilde{\omega}_1) p(\theta_2; \omega_2) \cdots p(\theta_n; \omega_n)$$

- and resulting marginal posterior distributions given by

$$p(\theta_j \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}}) \quad \text{for } j = 1, \ldots, n.$$

Suppose also for this example that we want to quantify how sensitive parameter θ_3 is to the specified change to the prior of θ_1.

5.2.2 Relative entropy of marginal posteriors

We begin by adopting a similar approach to Clarke and Gustafson [7] who used relative entropy to measure the difference between the two posterior distributions resulting from two different sets of inputs as in equation (5.1). However, we are interested in the relative entropy between two marginal posterior distributions resulting from two different sets of prior hyperparameters. More formally,

$$D\big(p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) \parallel p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})\big) = \int p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) log\left(\frac{p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})}{p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})}\right) d\theta_3. \tag{5.7}$$

The main complication is that we do not know the exact distributions $p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})$ or $p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})$. Instead we have a sample $\theta_3^{[1]}, \ldots, \theta_3^{[N]}$ from only one of them, namely $p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})$. To proceed we need to estimate these two densities which we do using Kernel density estimation methods.

5.2.3 Kernel density estimation

We can obtain an estimate of the baseline posterior $p(\theta_3 \mid \mathbf{x}, \omega)$ very easily using the sample $\theta_3^{[1]}, \ldots, \theta_3^{[N]}$ and the `density()` function in R. The theory is that if we have a sample $x_1, \ldots, x_n$ from a distribution D, then we can estimate its density using

$$\widehat{d}(t) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K\Big(\frac{t-x_i}{h}\Big) \approx E_d\Big[\frac{1}{h}K\Big(\frac{t-X}{h}\Big)\Big] \tag{5.8}$$

where K is the Kernel function and h is the bandwidth. An area of interest in the published literature has been how to find the optimal bandwidth for kernel density estimation an example of which is Sheather and Jones [46]. We use the R default value for the bandwidth which is calculated to be the standard deviation of the smoothing kernel.

Estimating the density of the 'changed' posterior $p(\theta_3 \mid \mathbf{x}, \widetilde{\omega})$ is slightly less straightforward than this as we do not have a sample directly from it. This is where importance sampling is useful.

5.2.4 Importance sampling

Importance sampling is a method of estimating an expectation with respect to one distribution using a sample from another distribution. Suppose that we are interested in estimating $E[r(X)]$ with respect to $f(x)$ but we only have a sample $x_1, \ldots, x_n$ from g(x). We can rewrite the expectation as follows

$$E_f[r(X)] = \int r(x)f(x)dx = \int \frac{r(x)f(x)}{g(x)}g(x)dx = E_g\left[\frac{r(X)f(X)}{g(X)}\right] \tag{5.9}$$

We can then use the sample from $g(x)$ to estimate

$$E_f[r(X)] \approx \frac{1}{n}\sum_{i=1}^{n} r(x_i)\frac{f(x_i)}{g(x_i)} \tag{5.10}$$

For our example, we need to find an expectation with respect to $p(\theta_3 \mid \mathbf{x}, \widetilde{\omega})$ but we only have the sample from $p(\theta_3 \mid \mathbf{x}, \omega)$. We can use equations (5.8)-(5.10) to write

$$\begin{aligned}
\widehat{p}(\theta_3^* \mid \mathbf{x}, \widetilde{\omega}) &\approx E_{p(\theta_3|\mathbf{x},\widetilde{\omega})}\left[\frac{1}{h}K\Big(\frac{\theta_3^* - \Theta_3}{h}\Big)\right] \\
&= E_{p(\theta_3|\mathbf{x},\omega)}\left[\frac{1}{h}K\Big(\frac{\theta_3^* - \Theta_3}{h}\Big)\frac{p(\Theta_3 \mid \mathbf{x}, \widetilde{\omega})}{p(\Theta_3 \mid \mathbf{x}, \omega)}\right] \\
&\approx \frac{1}{N}\sum_{i=1}^{N}\frac{1}{h}K\Big(\frac{\theta_3^* - \theta_3^{[i]}}{h}\Big)\frac{p(\theta_3^{[i]} \mid \mathbf{x}, \widetilde{\omega})}{p(\theta_3^{[i]} \mid \mathbf{x}, \omega)}
\end{aligned} \tag{5.11}$$

where θ_3^* is a particular value of θ_3 and we can now proceed using the sample we have. This is now in the form of weighted kernel density estimation equation.

5.2.5 Weighted Kernel density estimation

The Kernel density estimation equation given in (5.8) can be extended to include weights as follows. If we have a sample $x_1, \ldots, x_n$ from a distribution D, then we can estimate its density using

$$\widehat{d}(t) = \frac{1}{n}\sum_{i=1}^{n} w_i^* \frac{1}{h} K\Big(\frac{t - x_i}{h}\Big) \approx E_d\Big[w_i^* \frac{1}{h} K\Big(\frac{t - X}{h}\Big)\Big] \tag{5.12}$$

where w_i^* are normalised weights. It is clear that equation (5.11) is of this form with weights

$$w_i = \frac{p(\theta_3^{[i]} \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})}{p(\theta_3^{[i]} \mid \mathbf{x}, \boldsymbol{\omega})} = C\frac{p(\theta_1^{[i]}; \widetilde{\omega}_1)}{p(\theta_1^{[i]}; \omega_1)} \tag{5.13}$$

where C is an unknown constant. In order to ensure that $\int \widehat{p}(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})\ d\theta_3 \approx 1$ we must normalise the weights which also serves the purpose of eliminating the unknown C. Since the priors are known, we can proceed by using the `density()` function in R and normalised weights

$$w_i^* = \frac{p(\theta_1^{[i]}; \widetilde{\omega}_1)/p(\theta_1^{[i]}; \omega_1)}{\sum_{i=1}^{N} p(\theta_1^{[i]}; \widetilde{\omega}_1)/p(\theta_1^{[i]}; \omega_1)} \tag{5.14}$$

to calculate $\widehat{p}(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})$.

5.2.6 Numerical integration

We currently have an estimate of the densities $\widehat{p}(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})$ and $\widehat{p}(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})$ in the form of two sets of n coordinates which were output from R. In order to evaluate the right hand side of the relative entropy equation (5.7) we need to find

$$\int q(\theta_3)\ d\theta_3 \tag{5.15}$$

where

$$q(\theta_3) = \widehat{p}(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) log\left(\frac{\widehat{p}(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})}{\widehat{p}(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})}\right) \tag{5.16}$$

We can use the coordinates output from R to obtain a new set of $n + 1$ coordinates

$$(\{\theta_3\}_0, q_0), (\{\theta_3\}_1, q_1), \ldots, (\{\theta_3\}_n, q_n)$$

and estimate integral (5.15) using composite Simpson's rule

$$\int_a^b q(\theta_3)\ d\theta_3 \approx \frac{k}{3}\Big[q_0 + 2\sum_{j=1}^{\frac{n}{2}-1} q_{\{2j\}} + 4\sum_{j=1}^{\frac{n}{2}} q_{\{2j-1\}} + q_n\Big].$$

where $q_i = q(\{\theta_3\}_i)$, $\{\theta_3\}_i = a + ik$ and $k = (b-a)/n$ for $i = 0, \ldots, n$. We choose a and b such that $q(\theta_3) \approx 0$ for θ_3 outside $[a, b]$.

5.2.7 Summary

So far we have obtained a numerical value for how sensitive θ_3 is to a single change to a parameter in the prior of θ_1. This is essentially an estimate of the relative entropy between the baseline and 'changed' posteriors for θ_3. All that the method requires is knowledge of the baseline and changed prior densities for θ_1 and a single sample from each of the baseline posteriors θ_1 and θ_3.

Note that we have described the method in the context of this example but it is easy to generalise to obtain a numerical value for how sensitive any parameter θ_k is to a change in the prior of any parameter θ_j.

5.2.8 Graphical representation of sensitivity

In this section we consider the meaning of the numerical value for sensitivity that we have obtained. Since the quantity is essentially a measure of the 'difference' between the baseline and changed posterior, we compare it with a measure of the 'difference' between the baseline and changed prior in order to give us more of an insight into the relative size of the sensitivity value. We measure the difference between the priors again using the relative entropy metric to make the changes comparable. Figure 5.3 provides a way of picturing this. Suppose again that we change the prior of parameter θ_1 so that it becomes $p(\theta_1; \widetilde{\omega}_1)$ rather than $p(\theta_1; \omega_1)$. The x-axis shows the prior change

$$D\big(p(\theta_1; \omega_1) \parallel p(\theta_1; \widetilde{\omega}_1)\big)$$

and the y-axis shows

$$D\big(p(\theta_2 \mid \mathbf{x}, \boldsymbol{\omega}) \parallel p(\theta_2 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})\big)$$
$$D\big(p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) \parallel p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})\big)$$
$$\text{and} \quad D\big(p(\theta_4 \mid \mathbf{x}, \boldsymbol{\omega}) \parallel p(\theta_4 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})\big)$$

in the colours black, red and green respectively. As we increase the 'difference' between $p(\theta_1; \widetilde{\omega}_1)$ and $p(\theta_1; \omega_1)$ we see that each of the 'differences' between the marginal posteriors also increase. We can also see that θ_2 is the most sensitive to the prior change and is the only one for which the resulting posterior change is larger than the change made to the prior.

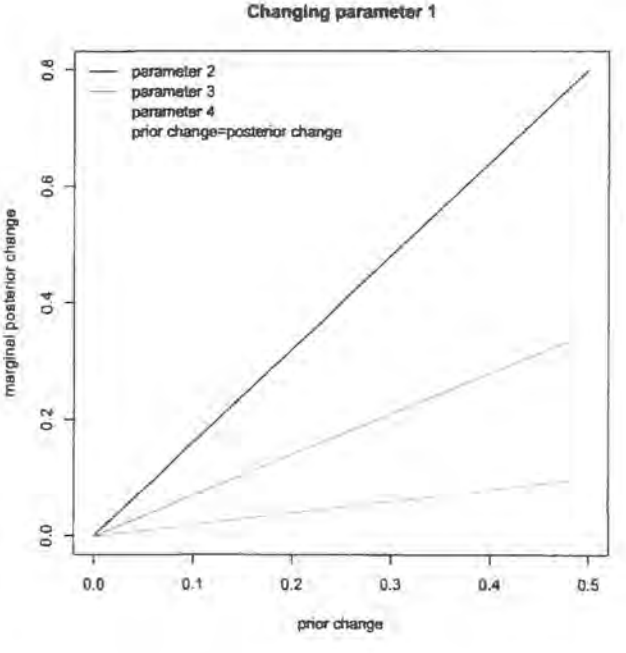


Figure 5.3: Prior change against marginal posterior change

5.3 How well does this method work?

In this section we consider how well the method works. In particular we want to know how accurately it estimates the true relative entropy between the baseline

and 'changed' marginal posteriors. In order to answer this question we simplify the problem to a situation where we are no longer thinking about the difference between two unknown posterior densities. Instead, we use the method to estimate the relative entropy between two standard distributions for which we know the true relative entropy. We will then be able to compare the estimates with the true relative entropy to draw meaningful conclusions about the accuracy of the method. We do this for two different sets of standard distributions: firstly measuring the relative entropy between two Normal densities and then between two Gamma densities. A second question of interest we bear in mind throughout this section is whether or not there is a prior change which is 'too big' for the method to work and causes it to break down. We will also look at how well importance sampling works in general, but since it is included in our method, we pay particular attention to the impact it has on the accuracy of the relative entropy estimate and also whether it can shed any light on the 'too big prior change' question. Again, we use a simplified version of our problem where we use importance sampling to estimate the density of a known Normal distribution using a sample from another known Normal distribution.

5.3.1 True and estimated relative entropy for a Normal distribution

Suppose we have two distributions $P_0 \sim N(0,1)$ and $P_1 \sim N(\mu, \tau^{-1})$ where μ is the mean and τ is the precision such that

$$\begin{aligned} p_0(x) &= \sqrt{\frac{1}{2\pi}} exp\{-\frac{1}{2}x^2\} \\ p_1(x) &= \sqrt{\frac{\tau}{2\pi}} exp\{-\frac{1}{2}\tau(x-\mu)^2\}. \end{aligned}$$

From equation (5.5) on page 108 we know that the true relative entropy is

$$D\Big(P_0 \| P_1\Big) = \frac{1}{2}\Big[\tau\mu^2 + \tau - log(\tau) - 1\Big]. \tag{5.17}$$

Suppose now that we take a sample $x_1, \ldots, x_n$ from P_0 and use the method described in section 5.2 to obtain an estimate of $D(P_0 \| P_1)$. First we use the `density()`

function in R to obtain a set of $(x, \widehat{p}_0(x))$ coordinates. We then use weights

$$w_i = \frac{p_1(x_i)/p_0(x_i)}{\sum_{i=1}^{N} p_1(x_i)/p_0(x_i)}$$

together with the `density()` function to obtain a set of corresponding coordinates $(x, \widehat{p}_1(x))$. Both sets of coordinates are then used to estimate $D(P_0 \| P_1)$ using Simpson's rule as in section 5.2.6.

We consider four different situations

- $P_0 \sim N(0,1)$ and $P_1 \sim N(\mu \geq 0, 1)$
- $P_0 \sim N(0,1)$ and $P_1 \sim N(\mu \leq 0, 1)$
- $P_0 \sim N(0,1)$ and $P_1 \sim N(0, \tau \geq 1)$
- $P_0 \sim N(0,1)$ and $P_1 \sim N(0, \tau \leq 1)$

and for each one we choose a value of the unknown μ or τ then find the true relative entropy using equation (5.17), along with 100 estimates of it using the method described in section 5.3.1. This is repeated for a number of different μ or τ values chosen to produce true relative entropys covering the range 0 to 5. The results are shown in Figure 5.4.

We can see from Figure 5.4 (a) and (b) that when only the mean between P_0 and P_1 differs, the estimate is very accurate up to a relative entropy of 1 and still fairly accurate up to a relative entropy of 5. This value of 5 corresponds to a difference in the means of approximately 3.2 which, in the context of changes to the prior, can be thought of as quite a big difference.

When it is the precision that differs between the two distributions, the method does not appear to perform so well. In (c) and (d) we see that things are only working well up to a relative entropy of around 0.5. We discuss in more detail in sections 5.3.3 and 5.3.4 why this might be the case but here we observe that a relative entropy of 0.5 could still be thought of as a sizeable change, for example see the red line in Figure 5.1(d).

So far we have only considered making changes to the prior of a Normal distribution but would the method still perform as well for another distribution? We now turn our attention to the Gamma distribution.

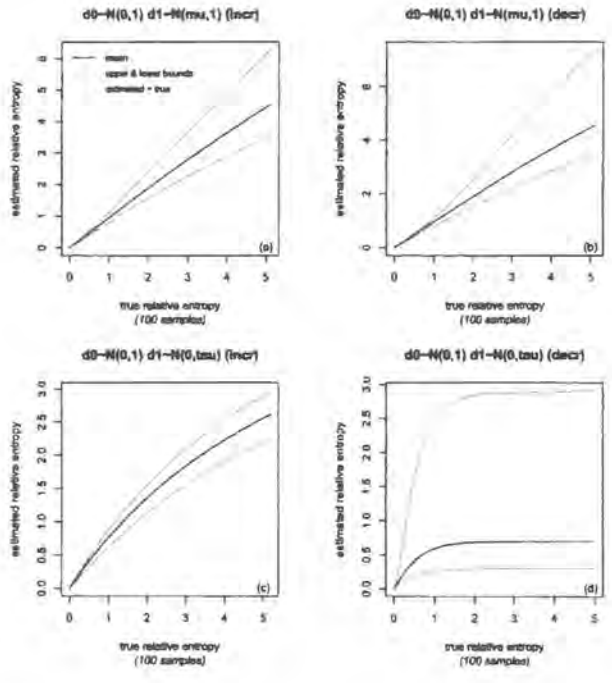


Figure 5.4: True versus estimated relative entropy for Normal distribution

5.3.2 True and estimated relative entropy for a Gamma distribution

Suppose now that we redefine P_0 and P_1 to have $Gamma(1,1)$ and $Gamma(a,b)$ distributions respectively, where a is the shape parameter and b is the rate such that

$$\begin{aligned} p_0(x) &= e^{-x} \\ p_1(x) &= \frac{b^a x^{a-1} e^{-bx}}{\Gamma(a)} \end{aligned}$$

Using equation (5.6) we know that the true relative entropy between P_0 and P_1 is

$$D\Big(P_0 \| P_1\Big) = \psi(1)(1-a) + b - 1 + log\Big(\frac{\Gamma(a)}{b^a}\Big) \tag{5.18}$$

and we consider the four situations

- $P_0 \sim Gamma(1,1)$ and $P_1 \sim Gamma(a \geq 1, 1)$
- $P_0 \sim Gamma(1,1)$ and $P_1 \sim Gamma(a \leq 1, 1)$
- $P_0 \sim Gamma(1,1)$ and $P_1 \sim Gamma(1, b \geq 1)$

- $P_0 \sim Gamma(1,1)$ and $P_1 \sim Gamma(1, b \leq 1)$

For each one we choose a number of values of the unknowns a or b then find the true relative entropy and 100 estimates of it for each value. The results are shown in Figure 5.5. We can see that the method doesn't seem to work as well for the

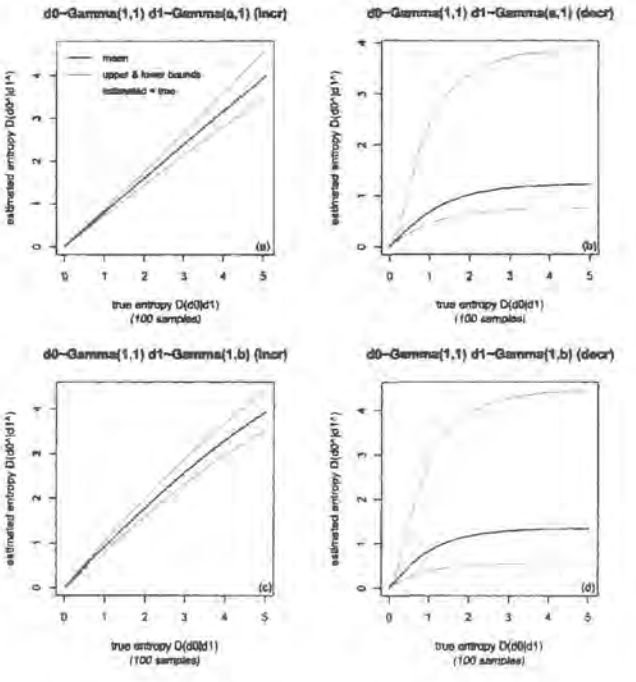


Figure 5.5: True versus estimated relative entropy for Gamma distribution

Gamma distribution as it does for the Normal distribution. The estimates all seem to underestimate the true relative entropy value. Having said that, they do seem to work quite well up to a relative entropy change of between 0.5 and 1 which could still be thought of as a sizeable change when we look at the red line in Figure 5.2.

5.3.3 How well does importance sampling work?

Recall from section 5.2.4 that we are interested in estimating $E[r(X)]$ with respect to $f(x)$ but we only have a sample $x_1, \ldots, x_n$ from g(x). Using equation (5.10) we can write

$$E_f[r(X)] \approx \frac{1}{n} \sum_{i=1}^{n} r(x_i) w(x_i) \tag{5.19}$$

where

$$w(x_i) = \frac{f(x_i)}{g(x_i)}$$

are the importance weights.

Much of the literature on analysing the method of importance sampling is concerned with how well it works *assuming* that it does work. However, the question still remains as to whether or not importance sampling works in the first place. In particular, are there any criteria that cause it to fail? Geweke [18] notes that for importance sampling to work $g(x)$ should closely mimic $f(x)$ and in particular that the tails of $g(x)$ should not decay faster than the tails of $f(x)$. Recall from section 5.2.4 that our method involves estimating an expectation with respect to one posterior (resulting from a changed prior) using only a sample from another posterior (resulting from a baseline prior). Intuition says that at some point there might be a prior change that is 'too big' in the sense that it produces a 'changed' posterior which no longer closely mimics the original and therefore would lead to a failure of importance sampling. However, so far we do not have a concrete way of showing whether or not this happens and if it does, at what sized prior change. One possible way of achieving this is to use the diagnostic described by Evans and Swartz [14] which is based on the weights and indicates whether or not importance sampling is working.

5.3.4 Importance sampling diagnostics

Evans and Swartz diagnostic

Suppose that $I = E_f[r(x)]$ and that $\widehat{I}$ is the approximation to this calculated using equation (5.19). Evans and Swartz [14] suggest that $\widehat{I}$ is a good approximation for I when n is large and that a good diagnostic for the failure of importance sampling is based on the coefficient of variation of $\widehat{I}$. They note that

$$\left(CV(\widehat{I})\right)^2 = \frac{1}{n}\left[n\sum_{i=1}^{n} w_*(x_i)^2 - 1\right] \tag{5.20}$$

where

$$w_*(x_i) = \frac{r(x_i)w(x_i)}{\sum_{i=1}^{n} r(x_i)w(x_i)}$$

is a measure of the significance of x_i in terms of its effect on $\widehat{I}$. They also note that since (5.20) is ≥ 0, it follows that

$$\frac{1}{n} \leq \sum_{i=1}^{n} w_*(x_i)^2 \leq 1$$

Furthermore $\sum_{i=1}^{n} w_*(x_i)^2 = 1$ if and only if one of the $w_*(x_i) = 1$ and the rest are 0. Therefore, they propose that a sensible diagnostic would be to compute $\sum_{i=1}^{n} w_*(x_i)^2$ and see if it is close to 1 as that would indicate a few large weights and therefore failure of importance sampling. Geweke [18] also endorsed this when he noted that bad behaviour exhibited by the estimated expectation can be as a result of large weights that turn up occasionally.

Simplified problem

In order to analyse the performance of the importance sampling part of our method, we consider a simplified version of our problem. We want to use importance sampling to estimate a density $f(x)$ using a sample $x_1, \ldots, x_n$ from density $g(x)$. Suppose for the purposes of this section that we know the true densities f and g.
Using equations (5.8) and (5.9) we can write

$$\widehat{f}(x^*) \approx E_f[r(X)] = E_g[r(X)w(X)]$$

where

$$r(X) = \frac{1}{h} K\Big(\frac{x^* - X}{h}\Big) \quad \text{and} \quad w(X) = \frac{f(X)}{g(X)}$$

and the Evans and Swartz diagnostic is

$$D_{ES} = \sum_{i=1}^{n} w_*(x_i)^2 = \sum_{i=1}^{n} \left(\frac{r(x_i)w(x_i)}{\sum_{j=1}^{n} r(x_j)w(x_j)} \right)^2.$$

Since $r(x)$ depends on x^*, it follows that we get a different value of D_{ES} for each value in the support of $f(x)$. Figure 5.6 shows two examples of importance sampling, one where it can be thought of as working well and the other where it doesn't work so well.

The plots show the true densities $g(x)$ and $f(x)$ via the black and blue lines respectively. The dashed blue line shows $\widehat{f}(x)$ obtained by importance sampling

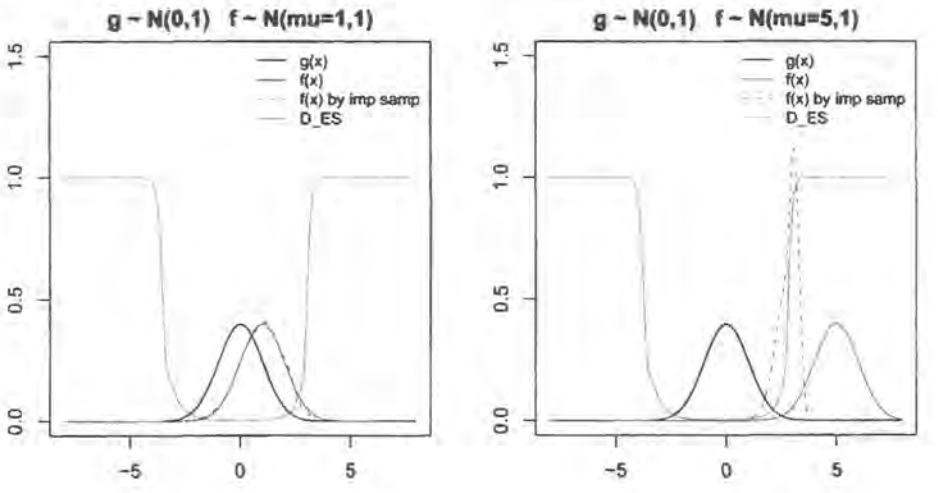


Figure 5.6: Importance sampling examples using the $\sum_{i=1}^{n} w_*(x_i)^2$ diagnostic

using the sample from $g(x)$. $g(x)$ is a Normal(0,1) density for both plots whereas $f(x)$ is a Normal(1,1) density for the first plot and Normal(5,1) for the second. The red line shows the diagnostic D_{ES} for values of $x \in (-8, 8)$. According to the diagnostic, importance sampling seems to fail for values of x that are out in the tails of $g(x)$. This suggests that if the majority of $f(x)$ is between the upper and lower tails of $g(x)$ then importance sampling will work. This is highlighted in the first plot. However, if the majority of $f(x)$ is beyond the tails of $g(x)$ as shown in the second plot, then importance sampling will fail. This is because for the second situation to work it would require importance sampling to sample values from $g(x)$ which aren't there. This example suggests that there is a prior change that would be 'too big' for the importance sampling part of the method to work. But it still remains to find out how big is 'too big'? To help provide further insight into this we consider another diagnostic.

Alternative diagnostic

Another importance sampling diagnostic could be

$$D_A = \left| \frac{\sum_{i=1}^{n} w(x_i)}{n} - 1 \right|$$

where a value close to 1 would indicate failure of importance sampling. This is because for $w(x_i) = \frac{f(x_i)}{g(x_i)}$ we have

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} w(x_i) = E_g\big[w(x_i)\big] = \int \frac{f(x_i)}{g(x_i)} g(x_i)dx_i = \int f(x_i)dx_i = 1$$

and therefore

$$\frac{\sum_{i=1}^{n} w(x_i)}{n} \to 1 \quad \text{as} \quad n \to \infty.$$

Suppose now that $g(x)$ is the baseline prior and $f(x)$ is the changed prior such that the weights are $w(x_i) = \frac{f(x_i)}{g(x_i)}$. We also assume here that $g(x)$ is a Normal(0,1) density. Figure 5.7 shows the value of this diagnostic D_A for different prior changes as measured by relative entropy covering the range 0 to 50.

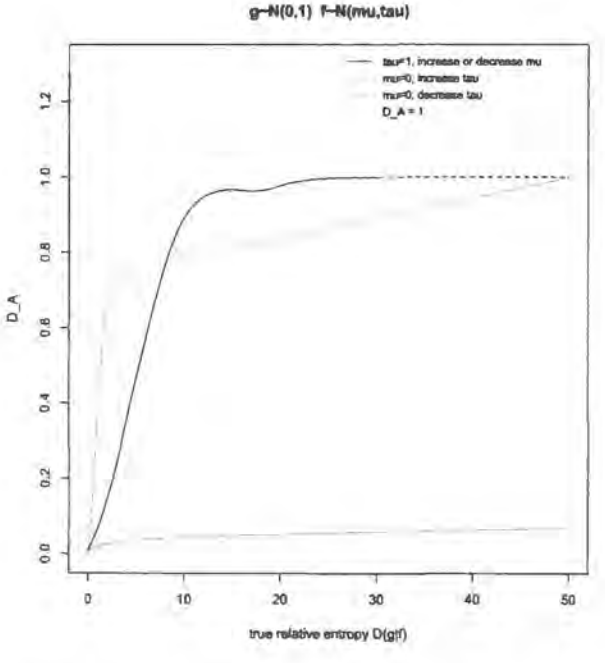


Figure 5.7: True relative entropy versus the $\left| \frac{\sum_{i=1}^{n} w(x_i)}{n} - 1 \right|$ diagnostic

The red line shows what happens if we keep the mean 0 and we increase the precision such that $f(x)$ is Normal(0, $\tau \geq 1$). As τ increases, the true relative entropy between g and f also increases. The green line shows the same but for decreasing τ such that $f(x)$ is Normal(0, $\tau \leq 1$). The black line shows what happens

if we keep the precision as 1 but shift the location either to the left or to the right, say for example that $f(x)$ is Normal($\mu \geq 0$, 1).

Ideally, for importance sampling to be working well, D_A should be close to 0. We can see that importance sampling seems to be working well up to a relative entropy of 50 when we increase the precision. This is because $f(x)$ is always within the tails of $g(x)$ so importance sampling is sampling values of $g(x)$ that are present, it is simply ignoring some. However in the reverse case, where we are decreasing the precision, we are forcing $f(x)$ to go beyond the tails of $g(x)$ and therefore requiring importance sampling to choose values from $g(x)$ which aren't there. We can see from the green line that the diagnostic is moving away from 0 towards 1 quite steeply almost immediately as we begin to increase the relative entropy. In this case a prior change of around 2 produces $D_A \approx 0.5$ which could be thought of as 'too big'. In the case where we change the mean, D_A again moves away from 0 relatively quickly as we increase the relative entropy. Here, a prior change of around 5 may be thought of as 'too big'.

5.3.5 Effect of importance sampling on the relative entropy estimate

So far we have considered what causes importance sampling to fail and also what size prior change may be thought of as 'too big' for the case where the prior in question is Normal. It would also be relatively easy to extend this to produce a plot like Figure 5.7 for other standard prior distributions. We now consider how the success or failure of importance sampling affects the relative entropy calculation.

We can note from section 5.2.6 that calculating the relative entropy estimate involves summing up a number of functions

$$q(x) = \widehat{g}(x) log\left(\frac{\widehat{g}(x)}{\widehat{f}(x)}\right) \tag{5.21}$$

which have been calculated for different values of x. In this section we have noted that our estimate of $\widehat{f}(x)$ is not good for values of x in the tails of $g(x)$. However it is clear from (5.21) that every time $\widehat{f}(x)$ appears in the relative entropy calculation, it is multiplied by $\widehat{g}(x)$. Since $\widehat{g}(x) \approx 0$ for values of x in the tails of $g(x)$, the values

of $\widehat{f}(x)$ which are not good are cancelled out. This means that it is possible for the importance sampling part of our method to fail but the relative entropy estimation still to be accurate.

5.3.6 Further exploration of statistical properties

In this section we consider how our method is affected by MCMC sample size as well as data size and mean. First we introduce a simple example. Suppose we have two prior distributions for θ

$$\begin{aligned} P_0 &\sim N(0,1) \\ P_1 &\sim N(\mu,\tau^{-1}) \end{aligned}$$

and the corresponding posteriors for θ given data $x_1,\ldots,x_n$ are

$$\begin{aligned} P_0(\theta \mid x_1,\ldots,x_n) &\sim N\Big(\frac{n\bar{x}}{1+n},\frac{1}{1+n}\Big) \\ P_1(\theta \mid x_1,\ldots,x_n) &\sim N\Big(\frac{\tau\mu+n\bar{x}}{\tau+n},\frac{1}{\tau+n}\Big). \end{aligned}$$

Using equation (5.5) on page 108 we know that the true relative entropy between the two priors is

$$D_{\text{prior}} = \frac{1}{2}\Big[\tau\mu^2+\tau-log(\tau)-1\Big]$$

and between the two posteriors is

$$D_{\text{post}} = \frac{1}{2}\Big[(\tau+n)\Big(\frac{\tau\mu+n\bar{x}}{\tau+n}-\frac{n\bar{x}}{1+n}\Big)^2+\frac{\tau+n}{1+n}+log\Big(\frac{1+n}{\tau+n}\Big)-1\Big].$$

If we take a sample $\theta_1,\ldots,\theta_N$ from $P_0(\theta \mid x_1,\ldots,x_n)$ we can then use the method of section 5.2 to obtain an estimate $\widehat{D}_{\text{post}}$ of the true posterior.

In order to see how the MCMC sample size affects the performance of our method we change the value of N and to see how the data affects it we change its size n and its mean $\bar{x}$. When data is introduced, the posterior will become less like the prior so it is interesting to see what effect this has on the performance of the method. We consider twelve different combinations of N,n and $\bar{x}$ (which are shown in Table 5.1) and for each we look at the effect of increasing or decreasing μ in the prior as well as increasing or decreasing τ. We obtain 100 of the estimates $\widehat{D}_{post}$ for each case and plot a summary of these against D_{prior}.

	N	n	$\bar{x}$
1	50	0	0
2	500	0	0
3	3000	0	0
4	3000	2	0
5	3000	30	0
6	3000	1000	0
7	3000	2	1
8	3000	30	1
9	3000	1000	1
10	3000	2	2
11	3000	30	2
12	3000	1000	2

Table 5.1: Twelve combinations of N, n and $\bar{x}$

MCMC sample size

The first three cases in Table 5.1 are looking at the effect of the MCMC size only. Since $n = 0$ it means that we have no data and the posteriors P_0 and P_1 are just the same as the priors. Note that we have already considered one example of this in section 5.3.1 on page 116 which uses sample size $N = 1000$. The plots for the first three cases are shown in Figures 5.8 to 5.10. As the sample size N increases we can see that the mean estimate line is getting closer to the true value and the upper and lower bounds are getting tighter around it. By the time we reach MCMC size of 3000 our method is working very well up to a prior change of around 3 for μ being changed but only up to a prior change of around 0.5 when τ is being changed. For further discussion of issues surrounding this, see the example in section 5.3.1.

Data size

For the next three combinations, we keep the sample size $N = 3000$ and data mean $\bar{x} = 0$ but change the data size. The plots for cases 4 to 6 are shown in Figures 5.11 to 5.13 respectively. We can see that there is an improvement in the accuracy of our

method for all of the prior changes as we move from no data to a small amount of data. For the changing μ cases, the upper and lower bounds become tighter around the mean and for the τ cases the mean line goes from being quite different from the true line to following it relatively closely. As we increase the data size from 2 to 30 another improvement in the method's performance is evident. For each of the prior change cases, the mean lines move closer to their true line and the bounds become tighter around them. However as we then increase the data size again to 1000, there doesn't appear to be any obvious change in the performance from that of the $n = 30$ case.

Data mean

We now consider moving the mean of the data away from the centre of the prior so that $\bar{x} = 1$ and then $\bar{x} = 2$ and again look at the effect these changes have on the method's performance. In particular we want to know if the performance is better or worse as $\bar{x}$ increases and does allowing $\bar{x} > 0$ have any effect on the increasing n behaviour we observed for $\bar{x} = 0$? We therefore introduce combinations 7 to 12 in Table 5.1 for which the summary plots can be seen in Figures 5.14 to 5.19 respectively.

We first consider the performance of the method as $\bar{x}$ increases. Looking at Figures 5.11, 5.14 and 5.17 we see, for $n = 2$, the effect of increasing $\bar{x}$ from 0 to 1 and then to 2. There is no obvious difference between them meaning that $\bar{x}$ has little effect for this data size. If we do the same for $n = 30$ (using Figures 5.12, 5.15 and 5.18) we see that there is little difference in the μs and, although the bounds change slightly for the τs, the mean line follows the true line just as closely in each of the three figures. This is also true for the $n = 1000$ case although there could be said to be a slight improvement in the performance of our method for the increasing τ case when $\bar{x} = 1$ or 2 than when $\bar{x} = 0$ (see the bottom left plot of Figures 5.13, 5.16 and 5.19).

We now consider the issue of whether introducing $\bar{x} > 0$ has any effect on the changing n behaviour we observed for cases 4 to 6. We see from Figures 5.14 to 5.19 that the pattern we observed is still the same when $\bar{x} = 1$. Increasing n from

30 to 1000 made no obvious difference when $\bar{x} = 0$ but when $\bar{x} = 2$ there is an improvement for $n = 1000$ in that the bounds in the increasing τ plot are tighter, but this is only slight.

Summary

Changing the MCMC sample size and the data size affects the performance of our method, but there is no obvious difference in performance as the data mean is changed. Increasing the MCMC sample size leads to a definite improvement in how well our method works for all prior changes. Introducing data brings further improvement, the most notable being when τ is changed in the prior. When there is no data it can only said to be working well up to a prior change of 0.5 but by the time the data size reaches 30 this changes to a prior change of 5.

5.3.7 Method as a screening measure

Since the method cannot be said to be accurate for any sized prior change, it may be better to think of the method as a good screening measure to indicate where there is a parameter which is sensitive to the change rather than saying exactly how sensitive it is. The method would still reduce the time needed to check sensitivity to the prior distribution as there would be no need to run the MCMC simulation again for each change to the prior. Instead, the method could be used as an indication of where sensitivity may be and then the simulation could be rerun for only the necessary changes if higher accuracy is required.

Figure 5.8: $N = 50, n = 0, \bar{x} = 0$

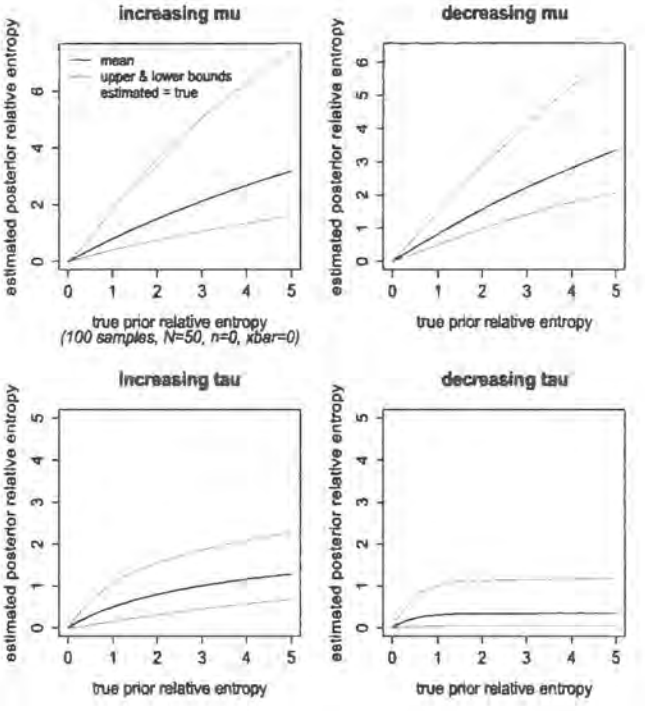


Figure 5.9: $N = 500, n = 0, \bar{x} = 0$

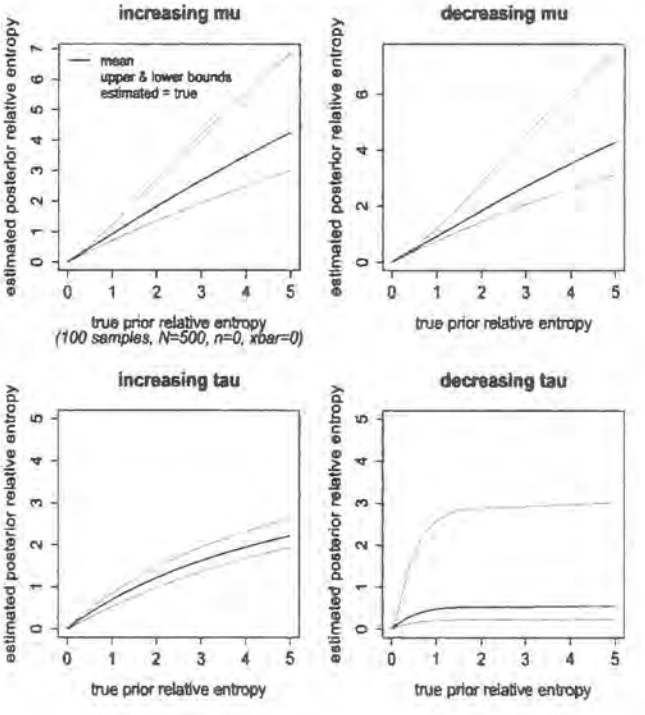


Figure 5.10: $N = 3000, n = 0, \bar{x} = 0$

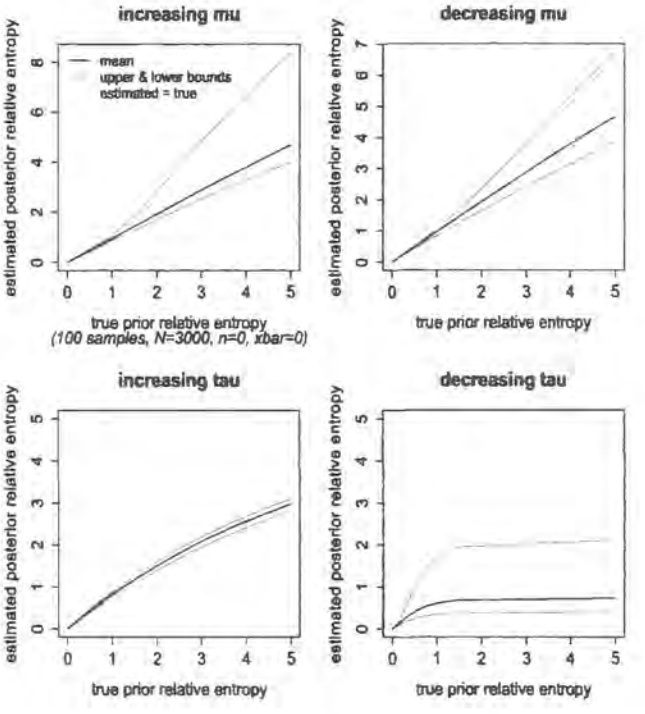


Figure 5.11: $N = 3000, n = 2, \bar{x} = 0$

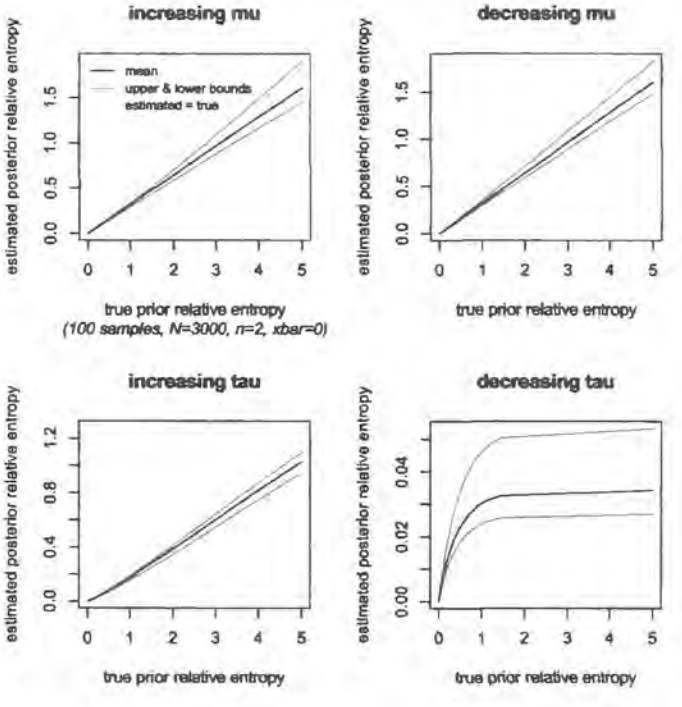


Figure 5.12: $N = 3000, n = 30, \bar{x} = 0$

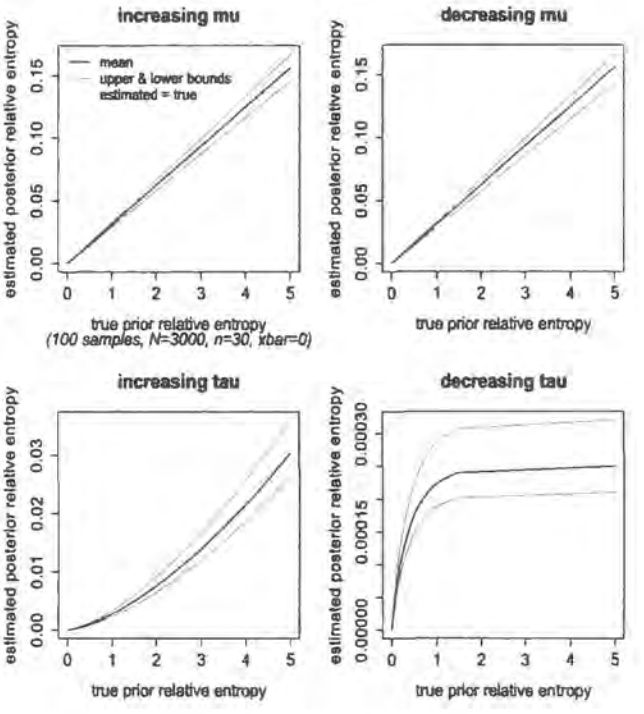


Figure 5.13: $N = 3000, n = 1000, \bar{x} = 0$

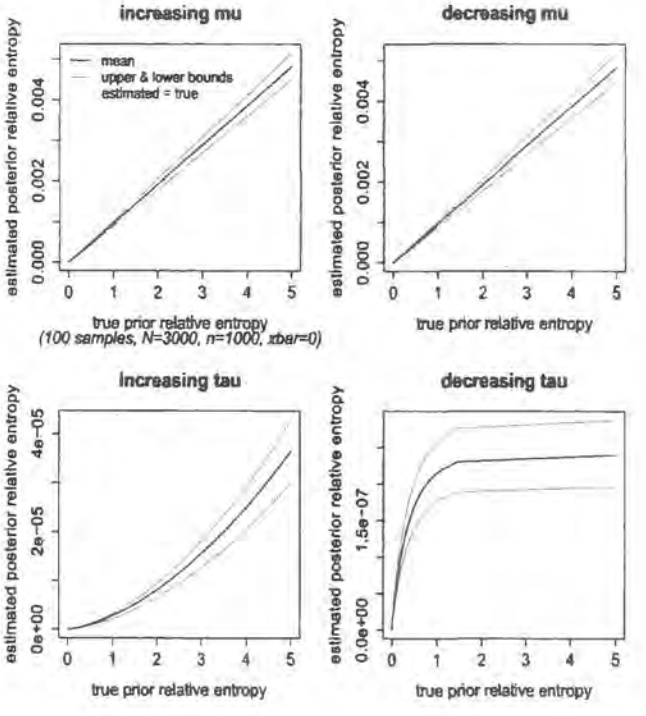


Figure 5.14: $N = 3000, n = 2, \bar{x} = 1$

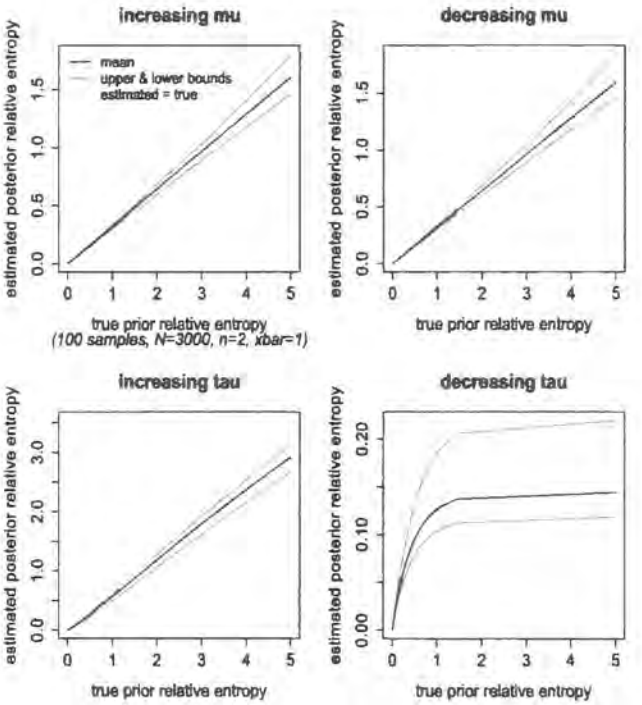


Figure 5.15: $N = 3000, n = 30, \bar{x} = 1$

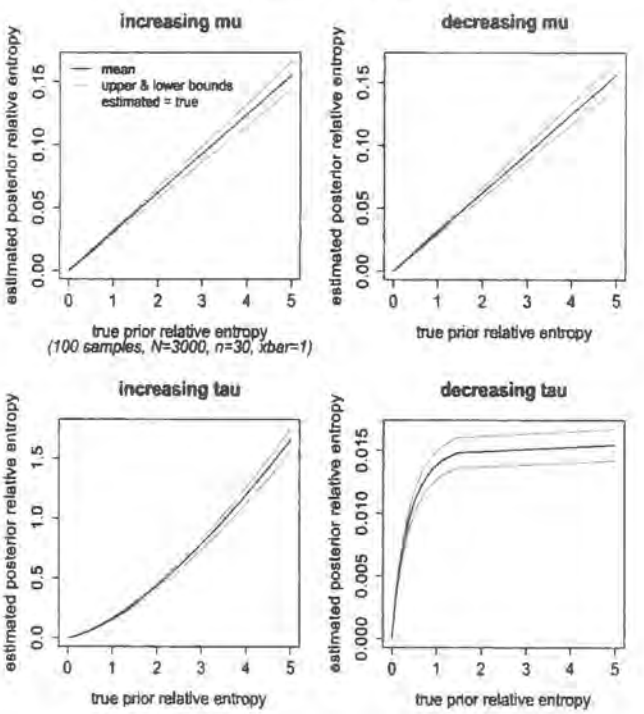


Figure 5.16: $N = 3000, n = 1000, \bar{x} = 1$

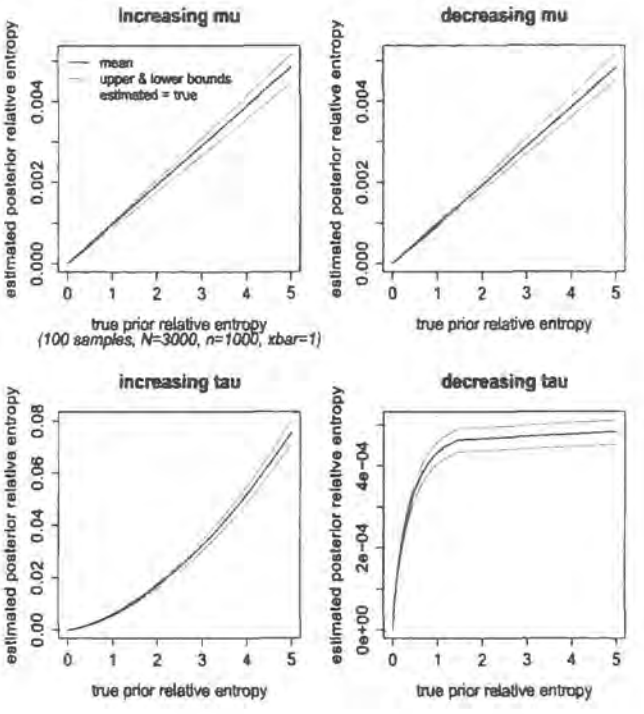


Figure 5.17: $N = 3000, n = 2, \bar{x} = 2$

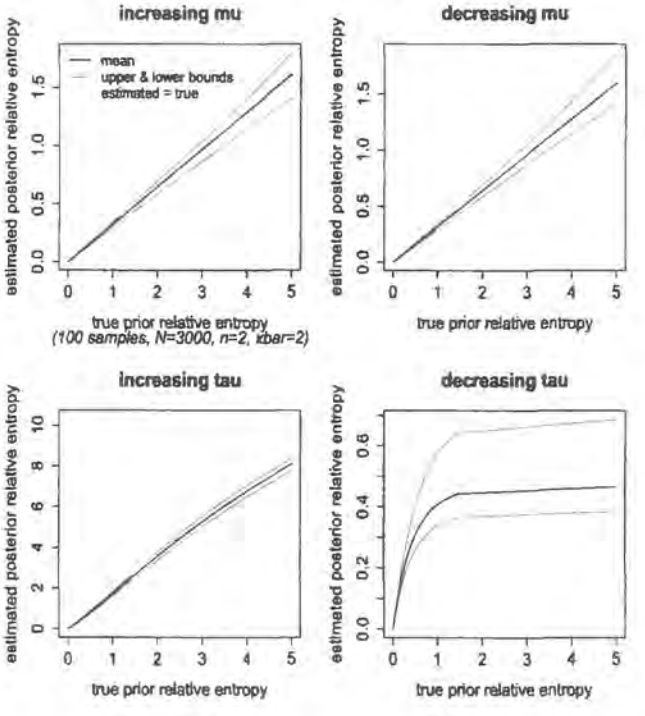


Figure 5.18: $N = 3000, n = 30, \bar{x} = 2$

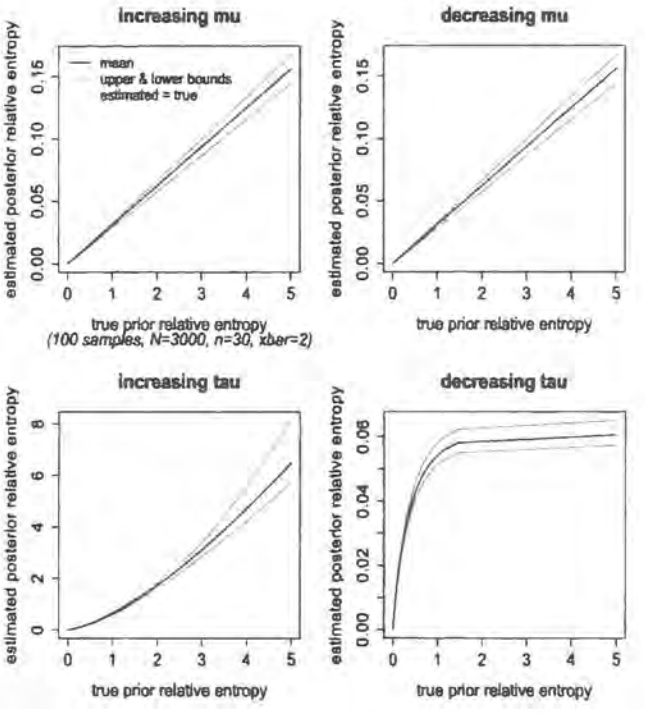
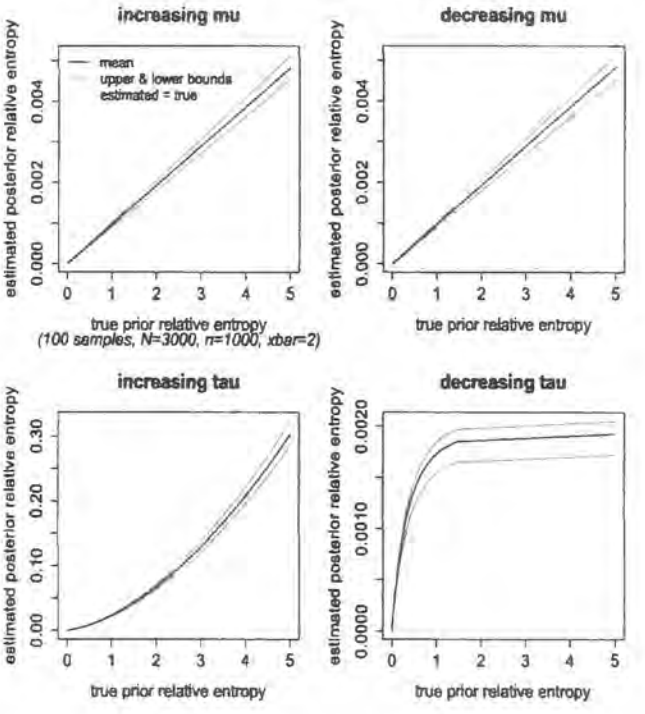


Figure 5.19: $N = 3000, n = 1000, \bar{x} = 2$



5.4 Replacing relative entropy with Kolmogorov distance

In this section we consider whether or not the method would perform any better if we used another metric instead of relative entropy to measure the difference between distributions. Although relative entropy is one of the most widely used metrics for such purposes, the tails of the distributions have a big influence on it. The Kolmogorov distance metric behaves differently to relative entropy in that the tails of the distributions involved are of less importance. Instead it measures the maximum distance between two distribution functions which can occur at any part of the distribution, not just in the tails. More formally,

$$D\Big(P\|Q\Big) = \sup_x \mid P(x) - Q(x) \mid, \quad x \in \mathbb{R}$$

which assumes values in $[0, 1]$. We can get an idea of how the Kolmogorov measure behaves for Normal and Gamma distributions from Figures 5.20 and 5.21 respectively. The changes highlighted in red, green and blue are comparable with those

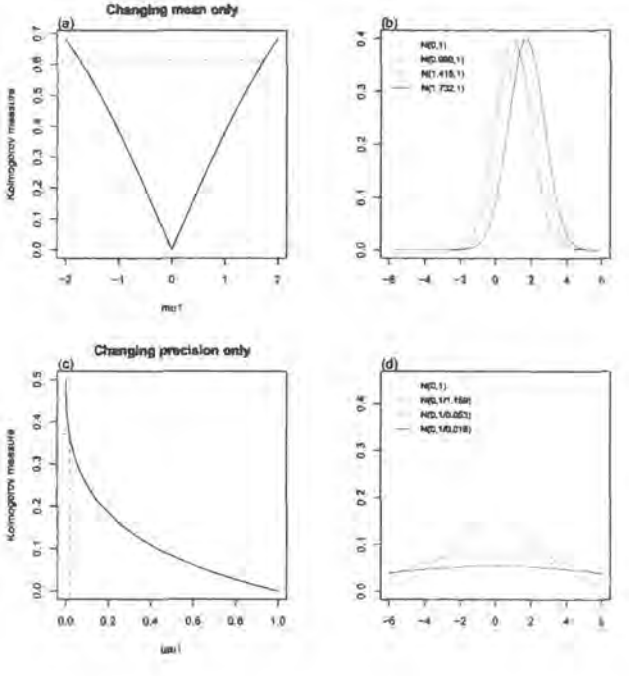


Figure 5.20: Changes to the Normal distribution measured by Kolmogorov distance

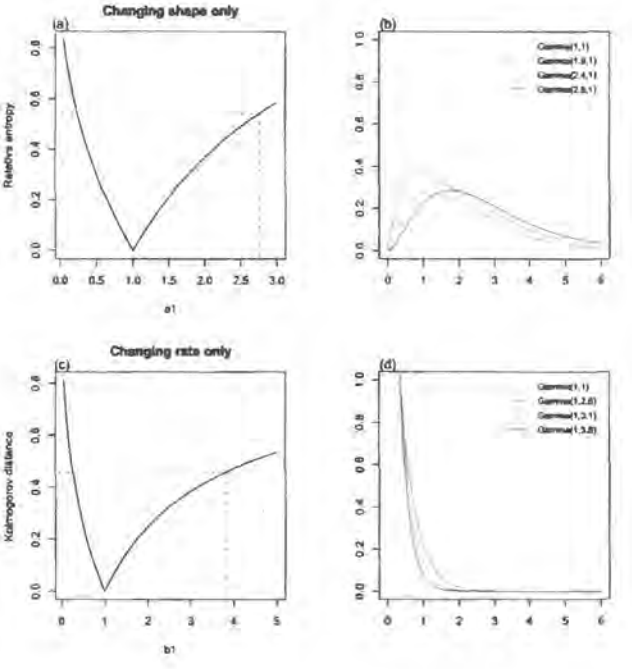


Figure 5.21: Changes to the Gamma distribution measured by Kolmogorov distance

highlighted in Figures 5.1 and 5.2 when we looked at the behaviour of relative entropy.

Figure 5.22 shows the relationship between relative entropy and the Kolmogorov distance. Although they all have the same curved shape, the scales are slightly different depending on the distribution and parameter change made.

5.4.1 Implementing the method

We saw in sections 5.2.3 to 5.2.5 how to obtain an estimate for $p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega})$ and $p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})$ in the form of two sets of coordinates. We now want to find an estimate of

$$D\Big(p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) \parallel p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})\Big) = \sup_t \left| \int_{-\infty}^{t} p(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) - p(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}}) \; d\theta_3 \right|.$$

For any value of $t \in \mathbb{R}$, we can estimate the above integral using these sets of coordinates and the numerical integration method outlined in section 5.2.6. However, in this case $q(\theta_3)$ defined in (5.16) becomes $\widehat{p}(\theta_3 \mid \mathbf{x}, \boldsymbol{\omega}) - \widehat{p}(\theta_3 \mid \mathbf{x}, \widetilde{\boldsymbol{\omega}})$ and b is taken

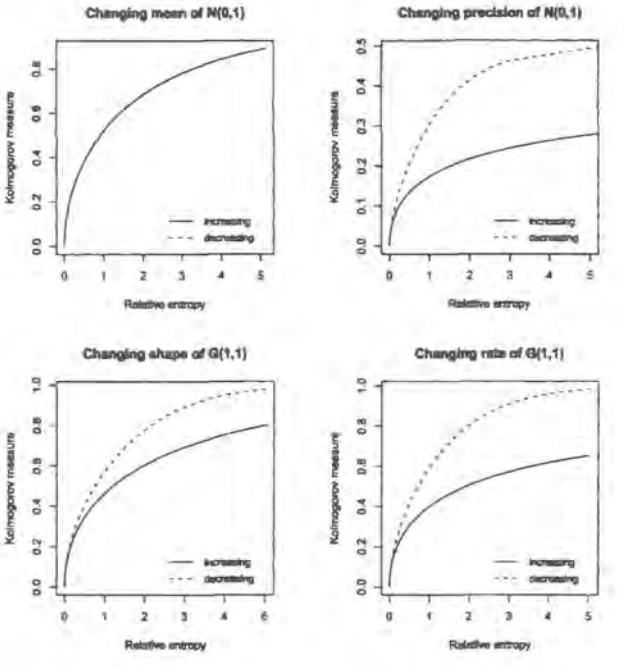


Figure 5.22: Relationship between relative entropy and Kolmogorov distance

to be equal to t. We then repeat this process for a number of different values of t and take the maximum of these to be our Kolmogorov distance measure.

5.4.2 How the method performs

In section 5.3 we looked at how well the relative entropy method performed for Normal and Gamma distributions. Here, we repeat this analysis but for the Kolmogorov distance method. In other words, we will compare the estimated Kolmogorov distance with the true one for the two different sets of distributions. Figures 5.23 and 5.24 show the true measure against the maximum, minimum and mean estimated values from 100 samples. These are for the Normal and Gamma distributions respectively. The x-axes cover the same parameter values that produced a relative entropy of 0 to 5 to make the plots comparable to those in Figures 5.4 and 5.5 of section 5.3.

In general, the method involving the Kolmogorov distance estimates the true distance better than the method involving relative entropy estimates the true relative

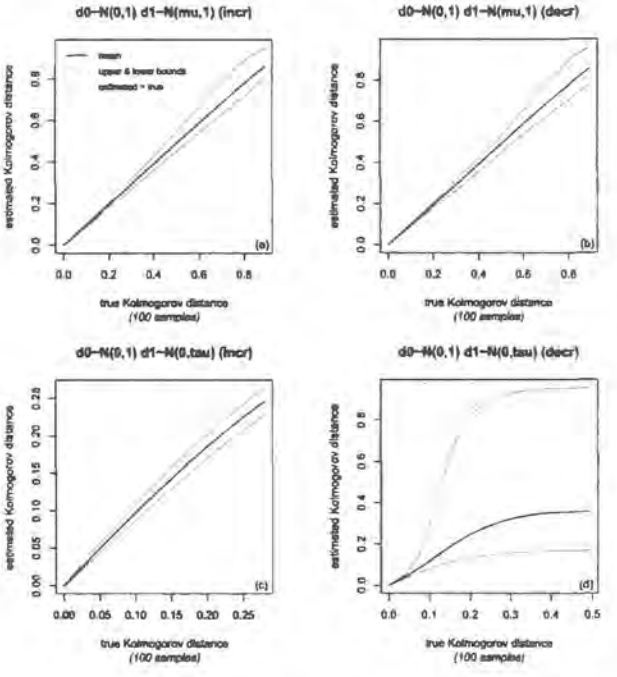


Figure 5.23: True versus estimated Kolmogorov distance for Normal distribution

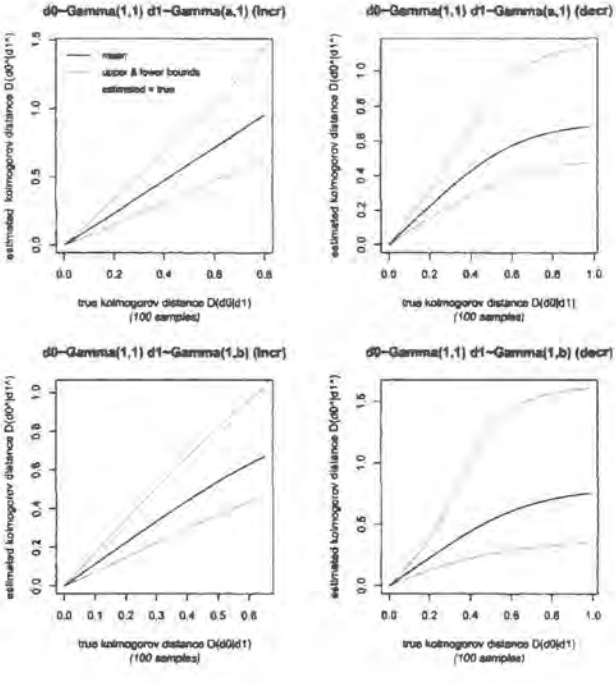


Figure 5.24: True versus estimated Kolmogorov distance for Gamma distribution

entropy.

5.4.3 Other metrics

Other metrics we could also consider include the Hellinger distance and the χ^2-distance. As noted by Gibbs and Su [19], they are regularly used along with relative entropy to quantify the distance between densities $p(x)$ and $q(x)$ from the same family indexed by different parameters. The Hellinger distance assumes values in $[0, \sqrt{2}]$ and is given by

$$D(P \parallel Q) = \Big[\int \Big(\sqrt{p(x)} - \sqrt{q(x)} \Big)^2 dx \Big]^{1/2}.$$

The χ^2-distance assumes values in $[0, \infty]$ and is given by

$$D(P \parallel Q) = \int \frac{\big(p(x) - q(x)\big)^2}{p(x)} dx.$$

However, it is likely that these metrics wouldn't work well for our method. This is because our method would involve estimating $q(x)$ using only a sample from $p(x)$ by importance sampling. As we noted in sections 5.3.3 and 5.3.4, importance sampling would fail for any $q(x)$ whose tails extended beyond those of $p(x)$. We noted in section 5.3.5 that this didn't become a probelm for relative entropy because every time $\widehat{q}(x)$ appeared, it was multiplied by $\widehat{p}(x) \approx 0$ thus cancelling out the poor estimate $\widehat{q}(x)$. However in the case of the above metrics, it seems that poor estimates for $q(x)$ wouldn't be cancelled out and therefore would negatively affect the result.

5.5 Application

In this section we apply the marginal sensitivity method described in section 5.2 to the complex Bayesian model of Mugglin *et al.* [36] using the space-time count data provided by NHS Direct. See sections 3.2 and 3.1 for details of the model and data respectively.

5.5.1 MCMC sample from baseline prior

Recall that the prior parameters and hyperparameters for the model are given by equations (3.4) to (3.7) on page 44. Since ϕ_{min} and ϕ_{max} are determined from the

eigenvalues of C we do not need to choose values for them. The hyperparameters we need to decide on are

$$\omega = (\mu_{\beta_0}, \tau_{\beta_0}, \mu_{\beta_1}, \tau_{\beta_1}, \mu_{\beta_2}, \tau_{\beta_2}, \mu_{\theta_0}, \tau_{\theta_0}, \mu_{\theta_1}, \tau_{\theta_1}, \mu_{\theta_2}, \tau_{\theta_2}, a, b) \tag{5.22}$$

We use the hyperparameter values suggested by Mugglin *et al.* [36] for the baseline prior. Specifically,

$$\begin{aligned} \beta_\ell &\sim Normal(0, \tau = 0.25), \quad \ell = 0, 1, 2 \\ \theta_\ell &\sim Normal(0, \tau = 0.25), \quad \ell = 0, 1, 2 \\ \sigma^{-2} &\sim Gamma(0.25, 2.5) \end{aligned}$$

and fit the model to the data provided by NHS Direct using `LinBUGS` as described in section 3.3. We then obtain an MCMC sample of size 10000 from the posterior distribution of each of the parameters.

5.5.2 Marginal sensitivity

Here we change each of the hyperparameters in (5.22) in turn and consider which of the parameters σ^{-2}, β_ℓ and θ_ℓ (for $\ell = 0, 1, 2$) are most sensitive to the change. Figures 5.25 to 5.31 show the results for each change individually including separate plots for whether the hyperparameter has been increased or decreased.

To explain one of them more fully, the top left plot in 5.25 shows what happens when the hyperparameter a is increased away from its baseline value. The x-axis and y-axis are equivalent to those in Figure 5.3. Each parameter being influenced is shown in a different colour.

When looking at each of the plots in Figures 5.25 to 5.31, we can see a single parameter which is clearly most influenced by the change. As we would expect, it is that parameter whose prior is being changed. However, in some of the cases even the posterior changes of the most influenced parameter are quite small. It would therefore be useful to compare the plots with each other in order to see which prior changes actually have a significant impact on the marginal posteriors.

Figure 5.32 shows a summary of the information in these plots for one particular sized prior change, namely 0.2. The x-axis shows each of the hyperparameters that

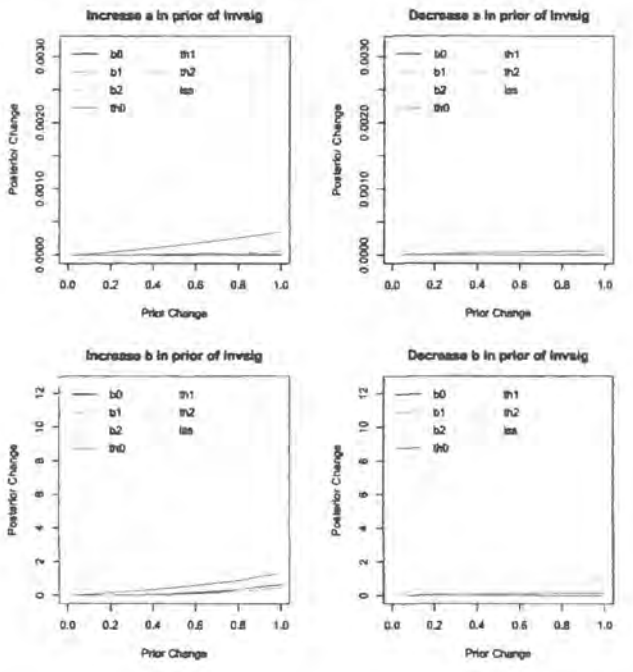


Figure 5.25: Marginal sensitivity to changes to hyperparameters a and b

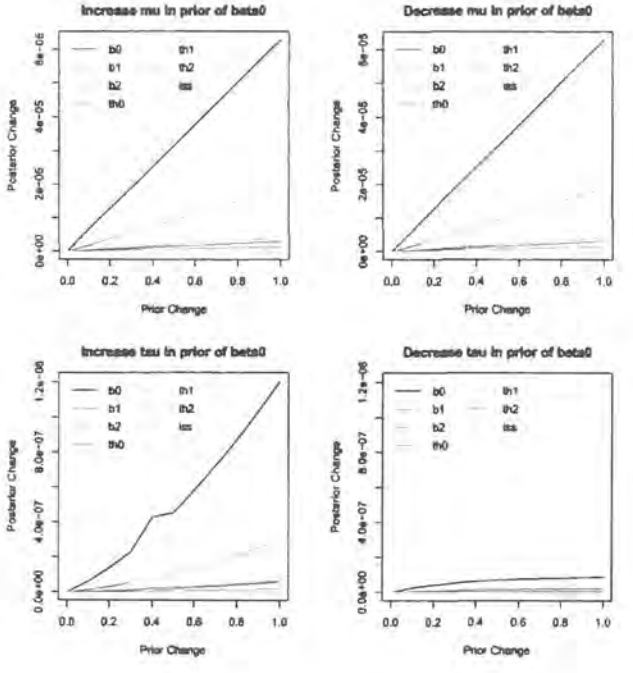


Figure 5.26: Marginal sensitivity to changes to hyperparameters μ_{β_0} and τ_{β_0}

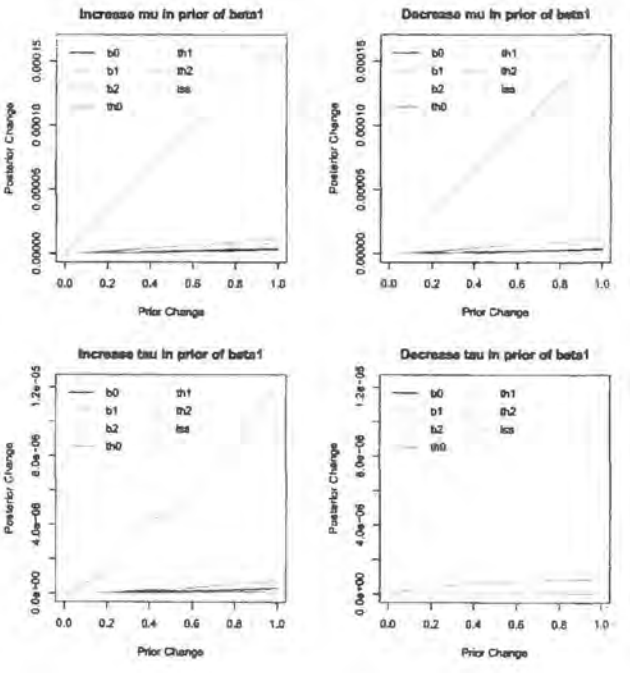


Figure 5.27: Marginal sensitivity to changes to hyperparameters μ_{β_1} and τ_{β_1}

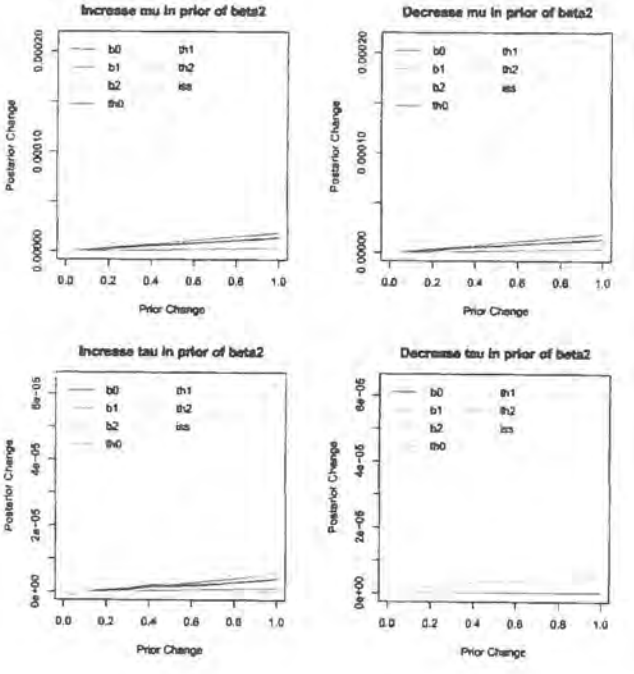


Figure 5.28: Marginal sensitivity to changes to hyperparameters μ_{β_2} and τ_{β_2}

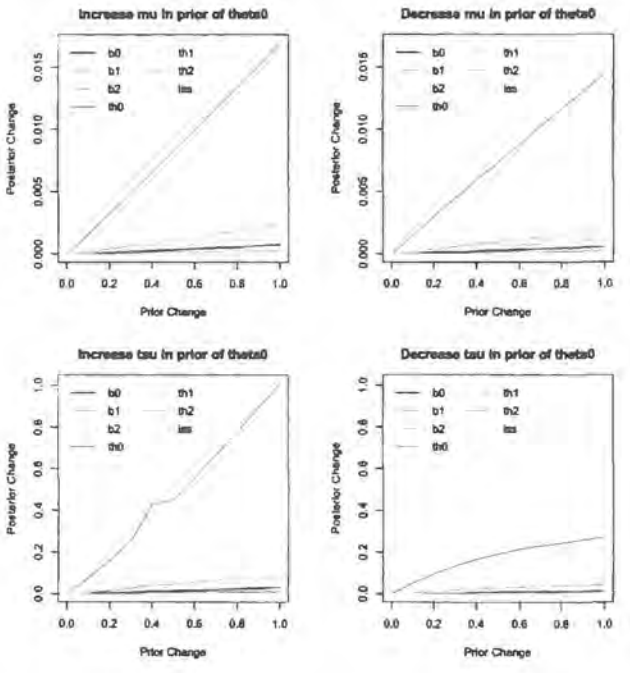


Figure 5.29: Marginal sensitivity to changes to hyperparameters μ_{θ_0} and τ_{θ_0}

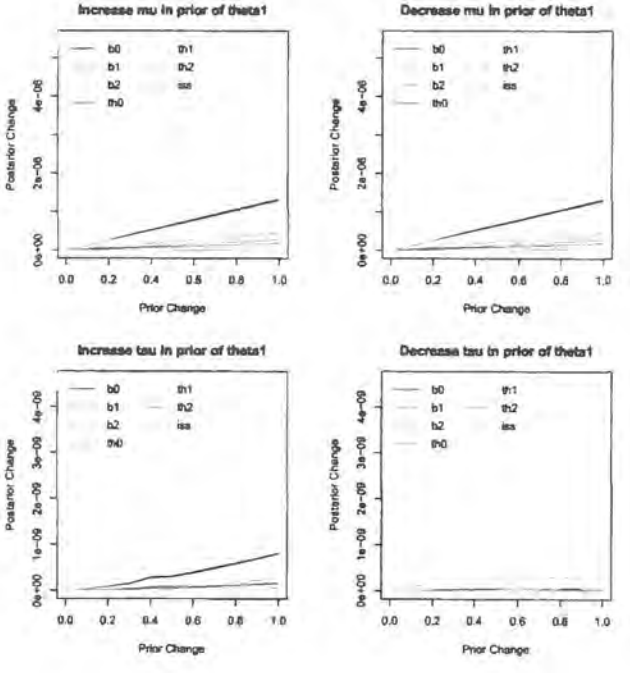


Figure 5.30: Marginal sensitivity to changes to hyperparameters μ_{θ_1} and τ_{θ_1}

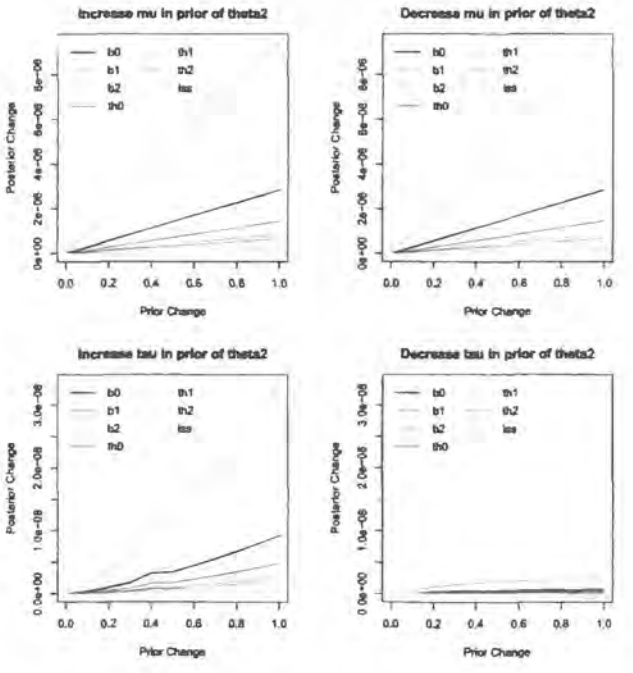


Figure 5.31: Marginal sensitivity to changes to hyperparameters μ_{θ_2} and τ_{θ_2}

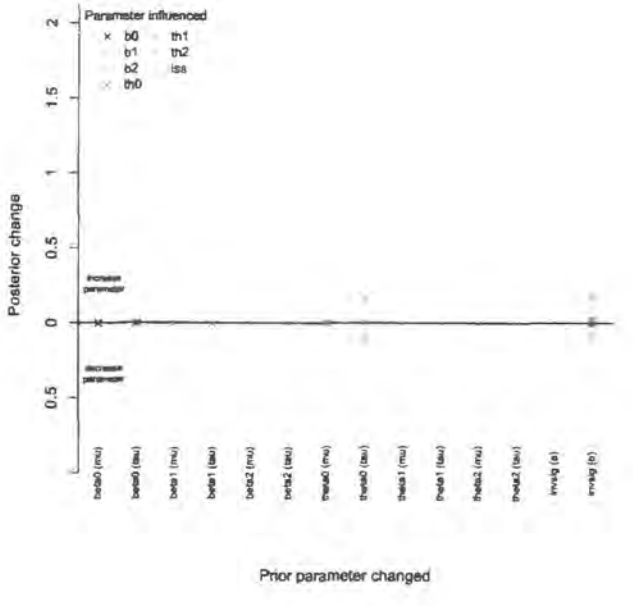


Figure 5.32: Summary of marginal sensitivity for prior change of 0.2

are changed and the y-axis shows the marginal posterior changes as in Figures 5.25 to 5.31. Above the zero line corresponds to when the hyperparameters are increased, and below to when they are decreased. The posterior change for each parameter is shown in a different colour.

We can see from this summary that the posterior is most sensitive to both increasing and decreasing the hyperparameter b, followed by τ_{θ_0} and the other changes have relatively little effect. We can therefore limit our attention to the detailed plots for these two hyperparameters only. In practise, producing a summary plot such as this first would be a good idea as it reduces the number of detailed plots we need.

Suppose now that we focus again on the method involving the Kolmogorov distance measure in place of relative entropy. Then instead of Figure 5.32, we get Figure 5.33. Although it is difficult to compare the two plots exactly due to different scales, it is still clear that they both agree that the posterior is most sensitive to changing the hyperparameter b, followed by changing τ_{θ_0}. It is also interesting to note that Figure 5.33 suggests that changing μ_{θ_0} has an impact on the posterior too, albeit relatively small compared to the effect of the other two.

5.5.3 Sensitivity of the full posterior

In this section we show that our marginal sensitivity method produces results that are consistent with the Clarke and Gustafson [7] 'full sensitivity' method (described in section 5.1.1) but that our results are more informative. Figure 5.34 shows a similar summary plot to 5.32, but this time showing the sensitivity of the full posterior distribution to the prior change using the method of Clarke and Gustafson [7]. In this case we want to find the relative entropy between two full posteriors $p(\boldsymbol{\vartheta} \mid \boldsymbol{x}; \boldsymbol{\omega})$ and $p(\boldsymbol{\vartheta} \mid \boldsymbol{x}; \widetilde{\boldsymbol{\omega}})$ given by

$$D\big(p(\boldsymbol{\vartheta} \mid \boldsymbol{x}; \boldsymbol{\omega}) \big\| p(\boldsymbol{\vartheta} \mid \boldsymbol{x}; \widetilde{\boldsymbol{\omega}})\big) = \frac{1}{2}(\widetilde{\boldsymbol{\omega}} - \boldsymbol{\omega})^T A_{PS}(\boldsymbol{\omega})(\widetilde{\boldsymbol{\omega}} - \boldsymbol{\omega}) \tag{5.23}$$

where

$$\begin{aligned}
\boldsymbol{\vartheta} &= (\beta_0, \beta_1, \beta_2, \theta_0, \theta_1, \theta_2, \sigma^{-2}), \\
\boldsymbol{\omega} &= (\mu_{\beta_0}, \tau_{\beta_0}, \mu_{\beta_1}, \tau_{\beta_1}, \mu_{\beta_2}, \tau_{\beta_2}, \mu_{\theta_0}, \tau_{\theta_0}, \mu_{\theta_1}, \tau_{\theta_1}, \mu_{\theta_2}, \tau_{\theta_2}, a, b) \\
&= (0, 0.25, 0, 0.25, 0, 0.25, 0, 0.25, 0, 0.25, 0, 0.25, 0.25, 2.5)
\end{aligned}$$

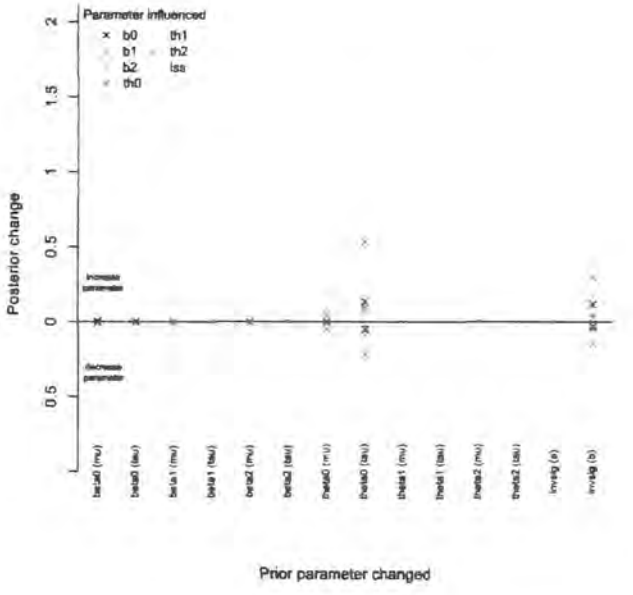


Figure 5.33: Summary of marginal sensitivity for prior change of 0.2 (using Kolmogorov distance)

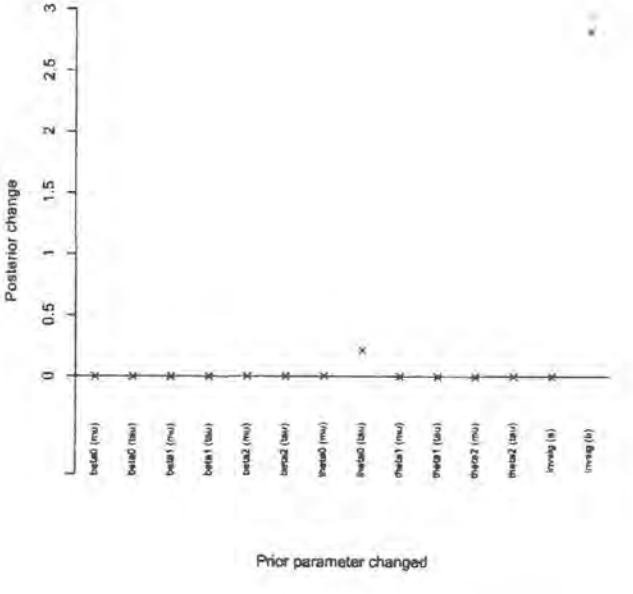


Figure 5.34: 'Full sensitivity' for prior change of 0.2

and $\widetilde{\omega}$ is as ω but with one of the hyperparameters changed by a relative entropy of 0.2. For example

$$\begin{aligned}\widetilde{\omega} &= (\widetilde{\mu}_{\beta_0}, \tau_{\beta_0}, \mu_{\beta_1}, \tau_{\beta_1}, \mu_{\beta_2}, \tau_{\beta_2}, \mu_{\theta_0}, \tau_{\theta_0}, \mu_{\theta_1}, \tau_{\theta_1}, \mu_{\theta_2}, \tau_{\theta_2}, a, b) \\ &= (0.6326, 0.25, 0, 0.25, 0, 0.25, 0, 0.25, 0, 0.25, 0, 0.25, 0.25, 2.5)\end{aligned}$$

We know from section 5.1.1 that

$$\left[A_{PS}(\boldsymbol{\omega})\right]_{ij} = \begin{cases} Var\left(\frac{\partial}{\partial\omega_i} logp(\boldsymbol{\vartheta};\boldsymbol{\omega})\right) & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

where $p(\boldsymbol{\vartheta};\boldsymbol{\omega})$ is the full prior, although since we are differentiating by ω_i we only need to concentrate on the part of the prior involving ω_i. Since all of the prior parts follow either a Normal or Gamma distribution, ω_i is one of the following:

1. The mean μ of a Normal distribution such that

$$Var\left(\frac{\partial}{\partial\mu} logp(\vartheta_i;\mu,\tau)\right) = \tau^2 Var(\vartheta_i)$$

2. The precision τ of a Normal distribution such that

$$Var\left(\frac{\partial}{\partial\tau} logp(\vartheta_i;\mu,\tau)\right) = \frac{1}{4}Var(\vartheta_i^2) + \mu^2 Var(\vartheta_i)$$

3. The shape a of a Gamma distribution such that

$$Var\left(\frac{\partial}{\partial a} logp(\vartheta_i;a,b)\right) = Var\left(log\vartheta_i\right)$$

4. The rate b of a Gamma distribution such that

$$Var\left(\frac{\partial}{\partial b} logp(\vartheta_i;a,b)\right) = Var(\vartheta_i).$$

These results, along with equation (5.23), can then be used to find the relative entropy for different $\widetilde{\omega}$ depending on which prior hyperparameter is changed. These values are shown in Figure 5.34 where the axes are equivalent to those described for Figure 5.32. Again this plot indicates that the posterior is most sensitive to changing b and τ_{θ_0} but does not tell us exactly which parts of the posterior are most affected as 5.32 does.

5.6 Marginal sensitivity analysis of BUGS output

So far we have seen the marginal sensitivity method being implemented for one set of BUGS output from one particular model. In this section we consider how easy it would be to produce a general piece of software for the marginal sensitivity analysis of BUGS output resulting from any model. We are not suggesting that the marginal sensitivity analysis be carried out in BUGS, but instead we consider what additional information we would need to know about the model given that we have some output from BUGS to analyse. We specifically look at how to pick this information out from a BUGS programmatic description of the model.

5.6.1 Necessary information

In order to implement our method as a general procedure we need to know the following information

- which parameters constitute the prior
- how the prior is specified for each parameter
- what should be changed in analysing sensitivity

Unfortunately the distinction between the prior and the likelihood is not always clear cut. One simple view of what constitutes the prior is that it is the distribution of nodes in the Directed Acyclic Graph (DAG) which have no parents. Assuming this is the case there are infinitely many ways to change the prior distributions but the most basic method is to change the prior-parameters while keeping the family the same. It is therefore possible to automate this process providing we can work out the DAG structure from the BUGS input file.

We now introduce programmatic descriptions for two of the models given in the WinBUGS examples[2] in order to illustrate how to extract the necessary information.

[2]available from `http://mathstat.helsinki.fi/openbugs/data/Examples/Volumei.html`

5.6.2 WinBUGS power plant pumps example

The BUGS language for the pumps example is as follows

```
model {
      for (i in 1 :  N) {
            theta[i] ~ dgamma(alpha, beta)
            lambda[i] <- theta[i] * t[i]
            x[i] ~ dpois(lambda[i])
      }
      alpha ~ dexp(1)
      beta ~ dgamma(0.1, 1.0)
}
```

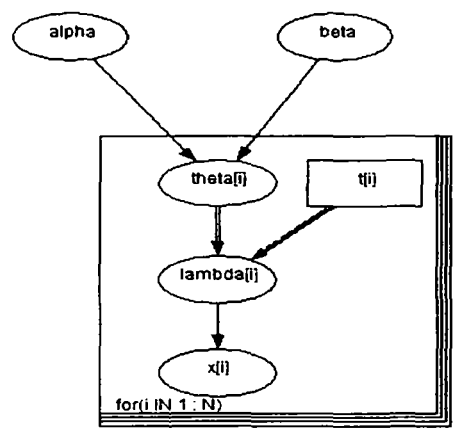


Figure 5.35: pumps DAG

and the corresponding DAG is given in Figure 5.35. We can see that the nodes which have no parents are `alpha` and `beta` and therefore constitute the prior. We can see from the model programmatic description that `alpha` $\sim$ `dexp(1)` and `beta` $\sim$ `dgamma(0.1, 1.0)` and therefore obtain the following information to be used in our sensitivity analysis

- the parameters are ϑ = (`alpha`,`beta`)

- alpha has an Exponential distribution with mean m and beta has a Gamma distribution with shape a and rate b
- the prior-parameters to be changed are $\omega = (m, a, b)$ with corresponding baseline values of $(1, 0.1, 1)$

We note here that it is possible to steer the marginal senstivity analysis by the way in which the prior is specified. For example, if the Gamma parameter beta was parameterised using mean and shape instead of the shape and rate then we could amend the BUGS file to read `beta ~ dgamma(shape, shape/mean)`. It would mean working a little harder to ensure that the distribution on beta was still a prior (with no random ancestors) but would naturally vary the shape and mean in the marginal sensitivity analysis. We now consider a second example to further illustrate how to extract the necessary information.

5.6.3 WinBUGS rats example

The BUGS language for the rats example is as follows

```
model {
    for (i in 1 :  N) {
        for (i in 1 :  N) {
            Y[i,j] ~ dnorm(mu[i,j], tau.c)
            mu[i,j] <- alpha[i]+beta[j]*(x[j]-xbar)
        }
        alpha[i] ~ dnorm(alpha.c,alpha.tau)
        beta[i] ~ dnorm(beta.c,beta.tau)
    }
    tau.c ~ dgamma(0.001,0.001)
    sigma <- 1 / sqrt(tau.c)
    alpha.c ~ dnorm(0.0,1.0E-6)
    alpha.tau ~ dgamma(0.001,0.001)
    beta.c ~ dnorm(0.0,1.0E-6)
    beta.tau ~ dgamma(0.001,0.001)
```

```
    alpha0 <- alpha.c - xbar * beta.c
}
```

and the corresponding DAG is shown in Figure 5.36. The nodes which have no parents are `alpha.tau`, `alpha.c`, `beta.c`, `beta.tau` and `tau.c` and therefore constitute the prior. Using the model programmatic description we can obtain the following information to input into our sensitivity analysis

- the parameters are ϑ = (`alpha.tau,alpha.c,beta.c,beta.tau,tau.c`)
- `alpha.tau`, `beta.tau` and `tau.c` have Gamma distributions with shapes $a_\alpha, a_\beta, a_\tau$ and rates $b_\alpha, b_\beta, b_\tau$ respectively. Furthermore, `alpha.c` and `beta.c` have Normal distributions with means μ_α, μ_β and precisions τ_α, τ_β respectively.
- the prior-parameters to be changed are

$$\omega = (a_\alpha, a_\beta, a_\tau, b_\alpha, b_\beta, b_\tau, \mu_\alpha, \mu_\beta, \tau_\alpha, \tau_\beta)$$

  with corresponding baseline values of

$$(0.0001, 0.0001, 0.0001, 0.0001, 0.0001, 0.0001, 0, 0, 1.0E-6, 1.0E-6).$$

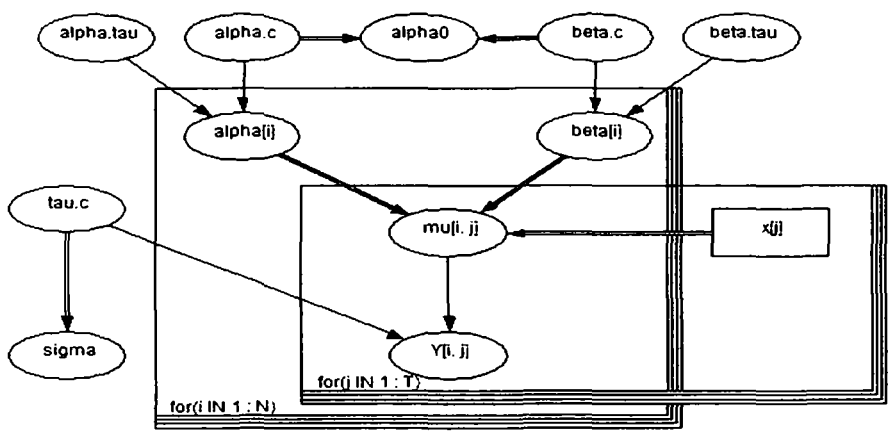


Figure 5.36: rats DAG

5.6.4 Summary

It is possible to produce a general piece of software to perform the marginal sensitivity analysis for any model which has BUGS output available. In order to do so it is necessary to know two things: the DAG structure of the model and it's BUGS programmatic description. Using the DAG we can pick out the prior by finding the nodes which have no parents. We then use the BUGS programmatic description to find out the details of the prior specification. More specifically we find out the distribution family of the prior and the baseline values for each prior-parameter which is to be changed. Any general piece of software produced would require some way of inputting this information.

Chapter 6

Conclusion

This thesis has been concerned with providing further statistical development in the area of space-time modelling with particular application to disease data. The first three chapters are essentially descriptive but chapter 3 does include the analysis of NHS data which has not been studied before. Chapters 4 and 5 introduce two methods which are new contributions to this area of research.

6.1 Analysis of DHF data

In chapter 2 we considered the method of empirical mode decomposition (EMD) as well as generalised linear modelling (GLM) to analyse the same data set consisting of cases of dengue haemorrhagic fever (DHF) in Thailand. EMD is not a statistical model but is purely descriptive. It isn't clear exactly how it works in that a number of people have produced different computer code for it and each of the methods differ slightly. In contrast, GLM is a widely used statistical model and therefore the estimates come with standard errors, residuals and formal statistical prodedures for comparing models. We can progressively increase the complexity of the model and check for improvement in fit, which is not possible with EMD. However, GLM is a simple type of statistical model and the most complex model we fitted still didn't eliminate all structure in the residuals.

6.2 Bayesian analysis of NHS data

In chapter 3 we analysed a space-time data set provided by NHS Direct which comprises the number of calls made to the north east site about the symptom cough. We adopted a Bayesian approach and analysed the data using the space-time hierarchical model of Mugglin *et al.* [36]. We found there to be a small degree of spatial structure in the spread of infection as well as a difference in the temporal patterns of relative risk between northern and southern regions. Those areas which were in the north of our study region generally had higher relative risk than those in the south.

However, a big question remains over how useful the data is and therefore how meaningful our conclusions actually are. The data only really captures a small proportion of the illness, for example the call rates were found to be low compared to GP consultation rates. In addition, due to reasons of confidentiality, the smallest level of spatial aggregation available to us was PCT. This was really too large to draw any meaningful conclusions about spatial structure. One example highlighting the problem is to do with neighbourhood structure in that almost every area ended up being classed as either a first- or second-order neighbour of every other area and this could negatively affect the results. If this data was going to be made more useful in future it should possibly be provided at postcode level.

6.3 Improved auxiliary sampling method

In chapter 4 we looked at improving the efficiency of MCMC for Poisson regression models. Such models involve at least one non-standard conditional distribution and in this chapter we provide a way to make it take the form of a multivariate Normal distribution by augmentation. It involves rearranging $p(\mathbf{x} \mid \mathbf{y})$ to comprise a multivariate Normal part and another part which can be approximated by one or more Normal mixture distributions. A sequence of latent variables is then introduced as the component indicators for these mixtures. This is then rearranged into another multivariate Normal distribution and is sampled from using an efficient block Gibbs sampling scheme.

This method improves upon recent work in this area. Applying the idea of Damien *et al.* [11] leads to a truncated version of the multivariate Normal distribution and generating from this is very difficult in comparison to generating from the distribution that our method produces. Frühwirth-Schnatter *et al.* [16] propose an auxiliary mixture sampling method which involves introducing two sequences of latent variables through data augmentation. Our method improves upon this by only requiring one sequence of latent variables. Further work in this area is by Rue *et al.* [45] who use a Gaussian approximation to the Poisson regression model. However, when the observed counts are small, the Poisson term is no longer approximated well by a Gaussian term so the method runs into problems. In contrast, our method does work well for small counts.

When comparing the posterior distribution obtained using auxiliary mixture sampling with that obtained from BUGS, they were seen to be very similar. Our method was tested further using many simulated data sets and the conclusion was reached that it works well for small counts but would need further mixture distributions to be added to make it work well for large counts.

Although our method is capable of producing a sample from the posterior equivalent to BUGS output, this chapter is essentially a proof of concept and no R package or other software has been developed for implementing it. This could be an area in which future research may be fruitful.

6.4 Marginal sensitivity method

The marginal sensitivity method is developed and tested in chapter 5. The method provides a way of quantifying how sensitive the posterior distribution of each parameter is to changes in the prior using just one set of MCMC output. It adds to current work in this area by adapting the idea of McCulloch [34] to work for MCMC output and also by allowing us to see exactly how changing each prior parameter affects the marginal posterior rather than the posterior as a whole.

The marginal sensitivity method involves using MCMC output and kernel density estimation to obtain the original posterior. It then uses the same MCMC output

along with a weighted form of kernel density estimation to obtain the posterior resulting from a changed prior. Simpson's rule is then used to estimate the relative entropy between these posteriors. The posterior changes can then be plotted allowing us to clearly see the effect that the prior change has on each of the marginal parameters.

The method was tested by running a number of different simulated examples and comparing the estimates to the true relative entropy. We first assumed that the prior was Normal with no data and looked at changing one of the parameters at a time. We found that when the mean was changed, the method worked well up to a prior change of 5 which can be thought of as a sizeable change. When the precision is changed, the method only works well up to a prior change of around 0.5. This is due to a failure in the importance sampling weights that were used in the weighted kernel density estimate. When the prior is Gamma with shape and rate parameters being changed one at a time, we find that the method works well to a prior change of 1 for all changes, which may still be thought of as a sizable change.

We also found that increasing the MCMC sample size and data size improves the performance of the method. When the MCMC size is 3000 and data size is 30, the method works well up to a prior change of 5 for both the mean and precision changes.

We then considered making an adaptation to the method so that the Kolmogorov distance measure was used in place of relative entropy. In general it appears that this estimates the true Kolmogorov distance slightly more accurately than the relative entropy method estimates the true relative entropy.

Since the method cannot be said to be accurate for any sized prior change, it may be better to think of it as a good screening measure to indicate where there is a parameter which is sensitive to the change rather than saying exactly how sensitive it is.

The final section of this chapter lays some ground work in what extra information we would need to know in order to produce a general piece of software to perform the marginal sensitivity analysis given any set of BUGS output. However, no software for the implementation of it has been produced as yet and future research would be

fruitful in this area. In addition to this, one could take this work further by studying the effect of changing more than one prior parameter at a time.

Bibliography

[1] D. Ashby (2006). Bayesian statistics in medicine: A 25 year review. *Statistics in Medicine* **25**, 3589–3631.

[2] M. Baker, G. E. Smith, D. L. Cooper, N. Q. Verlander, F. Chinemana, S. Cotterill, V. Hollyoak and R. Griffiths (2003). Early warning and NHS Direc: a role in community surveillance? *Journal of Public Health* **25**, 362–368.

[3] J. O. Berger, D. Rios Insua and F. Ruggeri (2000). Bayesian Robustness. In *Robust Bayesian Analysis*, eds D. Rios Insua and F. Ruggeri. Springer-Verlag, New York.

[4] S. P. Brooks (1998). Markov chain Monte carlo method and its application. *The Statistician* **47**(1), 69–100.

[5] S. Chib, F. Nardari and N. Shephard (2002). Markov chain Monte Carlo methods for stochastic volatility models. *Journal of Econometrics* **108**, 281–316.

[6] M. Chiogna and C. Gaetan (2004). Hierarchical space-time modelling of epidemic dynamics: an application to measles outbreaks. *Statistical methods and Applications* **13**, 53–69.

[7] B. Clarke and P. Gustafson (1998). On the overall sensitivity of the posterior distribution to its inputs. *Journal of Statistical Planning and Inference* **71**, 137–150.

[8] D. L. Cooper, E. Arnold, G. E. Smith, V. A. Hollyoak, F. Chinemana, M. Baker and S. J. O'Brian (2005). The effect of deprivation, age and gender on NHS Direct call rates. *British Journal of General Practice* **5**, 287–291.

[9] D. L. Cooper, N. Q. Verlander, G.E. Smith, A. Charlett, E. Gerard, L. Willocks and S. O'Brian (2006). Can syndromic surveillance data detect local outbreaks of communicable disease? A model using a historical cryptosporidiosis outbreak. *Epidemiology and Infection* **134**(1), 13–20.

[10] D. A. T. Cummings, R. A. Irizarry, N. E. Huang, T. P. Endy, A. Nisalak, K. Ungchusak and D. S. Burke (2004). Travelling waves in the occurrence of dengue haemorrhagic fever in Thailand. *Nature* **427**, 344–347.

[11] P. Damien, J. Wakefield and S. Walker (1999). Gibbs sampling for Bayesian non-conjugate and hierarchical models by using auxiliary variables. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **61**(2), 331–344.

[12] P. J. Diggle (2000). Overview of statistical methods for disease mapping and its relationship to cluster detection. In *Spatial Epidemiology: Methods and Applications*, eds P. Elliott, J. Wakefield, N. Best and D. Briggs. Oxford University Press.

[13] A. Doroshenko, D. Cooper, G. E. Smith, E. Gerard, F. Chinemana and N. Q. Verlander (2005). Evaluation of syndromic surveillance based on NHS Direct derived data in England and Wales. *Morbidity and Mortality Weekly Report* **54**(Suppl), 117–122.

[14] M. Evans and T. Swartz (2000). *Approximating integrals via Monte Carlo and deterministic methods*, eds A. C. Atkinson, J. B. Copas, D. A. Pierce, M. J. Schervish and D. M. Titterington. Oxford University Press.

[15] S. Frühwirth-Schnatter and R. Frühwirth (2007). Auxiliary mixture sampling with applications to logistic models. *Computational Statistics and Data Analysis* **51**(7), 3509–3528.

[16] S. Frühwirth-Schnatter, R. Frühwirth, L. Held and H. Rue (2007). Improved auxiliary mixture sampling for hierarchical models of non-Gaussian data. *Submitted to Journal of Computational and Graphical Statistics*

[17] S. Frühwirth-Schnatter and H. Wagner (2006). Auxiliary mixture sampling for parameter-driven models of time series of counts with applications to state space modelling. *Biometrika* **93**(4), 827–841.

[18] J. Geweke (1989). Bayesian inference in econometric models using Monte Carlo integration. *Econometrica* **57**(6), 1317–1339.

[19] A. L. Gibbs and F. E. Su (2002). On choosing and bounding probability metrics. *International Statistical Review* **70**(3), 419–435.

[20] W. R. Gilks, S. Richardson and D. J. Spiegelhalter, eds (1996). *Markov chain Monte Carlo in practice.* Chapman and Hall, London.

[21] S. Gschlöβl and C. Czado (2005). Does a Gibbs sampler approach to spatial Poisson regression models outperform a single site MH sampler. *Submitted*

[22] P. Gustafson (1996). Local sensitivity of inferences to prior marginals. *Journal of the American Statistical Association* **91**, 774–781.

[23] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N. Yen, C. C. Tung and H. H. Liu (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London Series A* **454**, 903–995.

[24] S. Kim, N. Shephard and S. Chib (1998). Stochastic volatility: Likelihood inference and comparison with ARCH models. *Review of Economic Studies* **65**, 361–393.

[25] L. Knorr-Held and J. Besag (1998). Modelling risk from a disease in time and space. *Statistics in Medicine* **17**(18), 2045–2060.

[26] L. Knorr-Held (2000). Bayesian modelling of inseparable space-time variation in disease risk. *Statistics in Medicine* **19**, 2555–2567.

[27] L. Knorr-Held and S. Richardson (2003). A hierarchical model for space-time surveillance data on meningococcal disease incidence. *Journal of the Royal Statistical Society Series C* **52**(2), 169–183.

[28] A. B. Lawson and M. Kulldorff (1999). A review of Cluster detection Methods. Chapter 7 in *Disease Mapping and Risk Assessment for Public Health*, eds A. B. Lawson, A. Biggeri, D Böhning, E. Lesaffre, J. Viel and R. Bertollini. Wiley, New York.

[29] A. B. Lawson (2001). *Statistical methods in spatial epidemiology.* Wiley, New York.

[30] A. B. Lawson and H. Zhou (2005). Spatial statistical modeling of disease outbreaks with paricular reference to the UK foot and mouth disease (FMD) epidemic of 2001. *Preventive Veterinary Medicine* **71**, 141–156.

[31] A. Le Menach, J. Legrand, R. F. Grais, C. Viboud, A. Valleron, A. Flahault (2005). Modeling spatial and temporal transmission of foot-and-mouth disease in France: identification of high-risk areas. *Veterinary Research* **36**, 699–712.

[32] D. J. Lunn, A. Thomas, N. Best, and D. Spiegelhalter (2000). WinBUGS – a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing* **10**, 325–337.

[33] K. V. Mardia, C. R. Goodall, E. Redfern and F. J. Alonso (1998). The Kriged Kalman filter (with discussion). *Test* **7**, 217-285.

[34] R. E. McCulloch (1989). Local model influence. *Journal of the American Statistical Association* **84**, 473–478.

[35] R. B. Millar (2004). Sensitivity of Bayes estimators to hyper-parameters with an application to maximum yield from fisheries. *Biometrics* **60**, 536–542.

[36] A. S. Mugglin, N. Cressie and I. Gemmel (2002). Hierarchical statistical modelling of influenza epidemic dynamics. *Statistics in Medicine* **21**, 2703–2721.

[37] A. Picado, F.J. Guitian and D.U. Pfeiffer (2007). Space-time interaction as an indicator of local spread during the 2001 FMD outbreak in the UK. *Preventive Veterinary Medicine* **79**, 3–19.

[38] A. E. Raftery and S. M. Lewis (1992). How many iterations in the Gibbs sampler? In *Bayesian Statistics 4*, eds J.M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith. Oxford University Press, pp 763–773.

[39] G. Rilling, P. Flandrin and P. Goncalves (2003). On empirical mode decomposition and its algorithms. In *Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing NSIP-03.*

[40] D. Rios Insua and F. Ruggeri, eds. (2000). *Robust Bayesian Analysis*, Springer-Verlag, New York.

[41] C. L. V. Rodeiro and A. B. Lawson (2006). Monitoring changes in spatio-temporal maps of disease. *Biometrical Journal* **48**(3), 463–480.

[42] G. Rodriguez-Yam, R. A. Davis and L. L. Scarf. Efficient Gibbs sampling of truncated multivariate normal with application to constrained linear regression. *Submitted.*

[43] H. Rue (2001). Fast sampling of Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B* **63**(2), 325–338.

[44] H. Rue and L. Held (2005). *Gaussian Markov Random Fields: Theory and Application.* Monographs on Statistics and Applied Probability, volume 104. Chapman & Hall, London.

[45] H. Rue, I. Steinsland and S. Erland (2004). Approximating hidden Gaussian Markov random fields. *Journal of the Royal Statistical Society, Series B* **66**(4), 877–892.

[46] S. J. Sheather and M. C. Jones (1991). A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society, Series B* **53**, 683–690.

[47] N. Shephard (1994). Partial non-Gaussian state space. *Biometrika* **81**(1), 115–131.

[48] G. E. Smith, D. L. Cooper, P. Loveridge, F. Chinemana, E. Gerard and N. Verlander (2006). A national syndromic surveillance system in England and Wales using calls to a telephone helpline. *Eurosurveillance Monthly* **11**(12), 220–224.

[49] A. F. M. Smith and A. E. Gelfand (1992). Bayesian statistics without tears: a sampling-resampling perspective. *The American Statistician* **46**, 84–88.

[50] L. A. Waller, B. P. Carlin, H Xia and A. E. Gelfand (1997). Hierarchical Spatio-Temporal Mapping of Disease Rates. *Journal of the American Statistical Association* **92**(438), 607–617.

[51] P. Yan and M. K. Clayton (2006). A cluster model for space-time disease counts. *Statistics in Medicine* **25**(5), 867–881.

Appendix A

NHS Direct Data

Table A.1: Number of cough calls in each PCT per week

	PCT											
Week	1	2	3	4	5	6	7	8	9	10	11	12
1	3	3	9	0	2	2	4	6	10	4	6	11
2	4	4	11	0	7	4	10	11	15	2	5	13
3	4	5	5	3	5	4	13	9	20	3	8	12
4	7	7	8	0	5	6	15	8	10	2	9	15
5	6	8	12	3	7	12	21	16	21	3	8	28
6	8	10	4	3	11	5	16	12	15	2	9	19
7	5	9	8	3	11	17	31	26	28	5	17	20
8	8	18	21	3	11	13	24	26	41	9	13	38
9	7	25	18	5	6	20	33	23	53	5	20	36
10	6	8	12	4	6	14	24	21	20	4	13	30
11	8	9	5	1	1	11	20	78	20	1	11	18
12	8	9	5	3	5	14	20	14	22	4	4	15
13	5	4	5	4	5	4	23	18	24	5	6	17
14	2	8	8	2	2	10	13	9	19	3	4	19
15	7	9	6	1	1	8	12	10	17	1	7	12
16	3	7	8	1	2	3	9	10	16	1	8	6
17	7	7	6	4	4	9	11	8	7	2	8	12

Table A.1 – continued from previous page

	PCT											
Week	1	2	3	4	5	6	7	8	9	10	11	12
18	4	1	4	0	1	10	12	5	9	2	5	7
19	3	2	10	2	0	8	9	10	17	2	5	18
20	1	10	4	1	4	7	11	11	9	2	7	16
21	5	4	6	1	5	11	13	16	17	1	3	20
22	3	8	4	3	3	9	16	11	23	6	6	14
23	2	9	7	1	3	5	11	10	18	1	4	9
24	2	3	5	0	3	5	9	5	9	4	3	13
25	5	1	1	0	1	8	7	2	9	2	7	9
26	1	5	5	2	3	6	12	9	8	1	3	10
27	1	6	4	1	6	8	15	8	11	2	9	6
28	1	3	5	1	3	2	3	2	9	0	2	8
29	3	0	8	0	4	6	10	10	10	0	6	7
30	0	0	5	0	1	5	7	4	10	1	3	13
31	3	2	5	0	0	2	9	3	14	1	5	8
32	2	3	1	1	1	3	10	7	10	0	4	7
33	1	2	1	1	1	2	5	4	1	2	0	5
34	3	3	2	4	1	4	6	4	7	3	1	5
35	3	2	2	1	0	3	4	5	3	2	2	6
36	0	4	3	1	0	5	7	1	6	1	2	2
37	1	2	2	0	1	5	6	4	3	1	1	9
38	1	4	1	0	2	2	3	4	5	0	4	5
39	3	3	3	2	0	1	6	3	6	1	4	1
40	1	1	4	1	0	2	3	1	4	2	3	0
41	1	4	1	1	0	2	5	5	3	0	1	7
42	1	2	4	0	0	1	7	4	11	0	1	5
43	1	0	3	0	0	2	5	4	5	0	6	11
44	2	2	4	0	2	3	2	1	5	1	4	4
45	1	0	1	0	1	1	0	2	2	0	0	1

Note that anyone wishing to use this data for publication should contact NHS Direct North East for permission.